From Criminology to Technology: The Birth of the Reasoning Substrate (RS)

How Behavioral Science Became the Foundation of a Deterministic Reasoning Architecture

---

Document Type: Technical White Paper

Version: 1.0

Classification: Internal Technical Review

Date: July 2026

Audience: Senior Engineers, Architects, Researchers, and Executives

Subject Areas: Substrate Architecture | Behavioral Science | AI Governance | Criminological Foundations

---

Table of Contents

1. Executive Summary 2

2. Introduction — Criminology as a Foundation for Structured Reasoning 3

3. Behavioral Science Foundations of RS 5

3.1 Routine Activity Theory (RAT) 5

3.2 Supervision Theory 6

3.3 CPTED — Crime Prevention Through Environmental Design 7

3.4 Structured Decision-Making Models 8

Table 1. Criminology Theory to RS Component Mapping 9

4. Determinism and the Rejection of Stochastic Reasoning 10

5. RS v1 as a Behavioral Model 11

Table 2. Criminology Concepts to HEG Component Mapping 13

6. RS v2 and Multi-Agent Behavioral Systems 14

7. Criminology and Injection Resistance 15

8. Criminology and Replayability / Auditability 16

9. Reasoning as State Evolution — The Philosophical Spine 17

Table 3. Criminology Concepts to RPU Component Mapping 18

10. Criminology vs. Computer Science as Origin Stories 19

11. Conclusion 21

12. References 22

SECTION 1

1. Executive Summary

The Reasoning Substrate (RS) was not built by constraining a language model. It was not built by adding guardrails to a probabilistic system and calling the result safe. RS was built by starting with a fundamentally different question — the same question criminologists have been asking for half a century: how do you supervise an actor you cannot fully trust? That question produced a substrate whose architecture reflects not the conventions of machine learning, but the deterministic, supervised, auditable logic of behavioral science. This paper establishes that lineage, maps it formally, and argues that criminology constitutes a superior intellectual foundation for safe AI substrate design than the computational traditions that currently dominate the field.

The core thesis is direct. RS's design is not analogous to criminological frameworks — it is derived from them. The RS cycle's operators correspond precisely to constructs that behavioral science has formalized over decades: state supervision, structured

decision-making, environmental constraint, threshold-based termination, and drift prevention. Each design choice in RS can be traced to a specific theoretical precedent. Drift theory (Matza, 1964) explains why RS requires deterministic state progression — unconstrained state evolution in a reasoning system fails by exactly the same mechanism that unconstrained behavior fails in criminological models. Routine Activity Theory (Cohen & Felson, 1979) provides the environmental structuring principle that governs how RS constructs and bounds its forward projection space. Supervision theory (Petersilia, 2003) provides the logic of threshold-based termination and audit-grade state logging. The Risk-Need-Responsivity model (Andrews & Bonta, 2010) provides the deterministic selection logic that governs the APPLY operator. None of this is metaphor. It is formal mapping.

The criminological lineage of RS explains four properties that matter most for safety and governance. First, determinism: because RS inherits from supervision frameworks that require binary, auditable state transitions, its cycle is non-stochastic by design. The same input, the same state, and the same COLLAPSE criteria will always produce the same outcome. Second, injection resistance: RS's substrate-above-prompt architecture is a direct translation of the principle that a supervision contract cannot be rewritten by the supervised individual's verbal claims. Adversarial inputs supply data to the substrate; they do not rewrite it. Third, auditability: because RS's state evolution is deterministic and fully logged, any reasoning trajectory can be exactly replayed — a requirement inherited from criminological supervision's mandate that every state transition be reviewable for legal and operational accountability. Fourth, termination safety: RS exits on pre-defined, substrate-enforced thresholds, not on self-determined completion signals — a design that mirrors supervision's violation-triggered termination logic.

The Human Expression Gateway (HEG) and the Reasoning Processing Unit (RPU) extend this foundation coherently. HEG applies the same criminological logic at the intake layer: environmental structuring governs what inputs are admissible; the capable guardian principle governs intake validation; structured decision-making governs intent classification. RPU applies the same logic at the optimization layer: situational crime prevention's cost-benefit hardening maps to RPU's energy landscape; supervision threshold logic maps to RPU's constraint satisfaction layer;

structured decision-making maps to RPU's deterministic plan scoring. The architecture is coherent from edge to core because its intellectual foundation is coherent.

This paper is organized as follows. Section 2 frames the relevance of criminology to AI substrate design. Section 3 establishes the formal mappings between behavioral science theory and RS components in four subsections, followed by Table 1. Sections 4 and 5 treat determinism and the RS v1 cycle in technical detail, followed by Table 2 mapping criminological concepts to HEG components. Section 6 extends the analysis to RS v2's multi-agent architecture. Sections 7 and 8 treat injection resistance and auditability respectively. Section 9 articulates the philosophical foundation, followed by Table 3 mapping criminological concepts to RPU components. Section 10 compares criminology and computer science as origin stories for substrate design. Section 11 concludes.

SECTION 2

2. Introduction — Criminology as a Foundation for Structured Reasoning

Criminology is not an obvious starting point for a reasoning architecture. The field is concerned with the causes, patterns, and prevention of criminal behavior — a domain that appears to have little in common with the design of AI substrate systems. That appearance is misleading. Beneath the surface, criminology and AI substrate design share a structural problem: both fields must answer the question of how a formal system maintains governed state evolution under adversarial pressure, across extended time horizons, without recourse to full trust in the actor being supervised.

The criminological literature has spent decades developing rigorous answers to that problem. By the 1960s, David Matza had formalized drift theory — the observation that delinquent behavior does not require a committed criminal identity but rather a weakening of the normative constraints that ordinarily bind behavior to legitimate channels (Matza, 1964). An individual who drifts is not choosing crime; they are

moving through an unstructured space that offers insufficient resistance to illegitimate trajectories. The parallel to AI reasoning failure is exact. A language model that hallucinates is not malfunctioning in the sense of a hardware fault — it is drifting through a stochastic space that offers insufficient structural resistance to incoherent trajectories. Drift in criminology and drift in AI reasoning are the same phenomenon at different scales: the absence of binding normative structure.

By the late 1970s, Lawrence Cohen and Marcus Felson had formalized Routine Activity Theory — the observation that criminal events require the convergence of three elements: a motivated offender, a suitable target, and the absence of a capable guardian (Cohen & Felson, 1979). The theory's analytical power lies not in the offender but in the environment. Structure the environment to eliminate the convergence, and the event does not occur. This is an environmental structuring principle applied to behavioral governance — and it maps with precision onto how RS constructs and constrains the forward projection space that determines what reasoning trajectories are reachable at any given cycle.

The supervision literature formalized the operational logic of behavioral governance. Petersilia's (2003) analysis of parole and prisoner reentry established that effective supervision requires scheduled state monitoring, explicit behavioral conditions, threshold-triggered termination, and full auditability of all transitions. These are not aspirational principles — they are operational requirements without which supervision fails. The same requirements apply to any system that claims to govern reasoning. A reasoning substrate that cannot log its state transitions, cannot define its termination conditions, and cannot replay its decisions for audit is not a safe substrate — it is supervision in name only.

RS, HEG, and RPU were designed in response to exactly these requirements. They were not designed to mimic the brain, to approximate human reasoning, or to maximize performance on benchmark tasks. They were designed to mirror the supervised, bounded, auditable logic of criminological behavioral frameworks — because that framework has been tested against adversarial actors, refined across decades of empirical research, and validated in operational environments where failure has serious consequences. The claim of this paper is not that criminology is metaphorically useful for AI design. The claim is that criminology's formal constructs

directly instantiate in RS's architecture, and that this lineage explains RS's most important safety properties.

Readers who are familiar with AI safety literature will note that the dominant paradigm in that field is alignment — the project of ensuring that AI systems pursue goals aligned with human values. RS does not contest the importance of alignment. But alignment is a property of a model, and a model that is not governed by a substrate is a model that is only as safe as its training allows. Substrate governance is not a replacement for alignment — it is a prerequisite for alignment to matter. Criminology understood this distinction before AI safety discourse existed. Supervision is not rehabilitation — it is the structural precondition that makes rehabilitation possible. RS is the engineering instantiation of that insight.

SECTION 3

3. Behavioral Science Foundations of RS

RS's design choices are not arbitrary. Each operator in the RS cycle corresponds to a behavioral science construct that has been independently formalized, empirically validated, and operationally tested in the criminological and behavioral science literature. The following four subsections establish those correspondences formally, proceeding from environmental structuring theory through supervision theory, situational crime prevention, and structured decision-making. The correspondences are not suggestive — they are the design record.

3.1 Routine Activity Theory (RAT)

Cohen and Felson (1979) introduced Routine Activity Theory to explain the conditions under which criminal events occur. The theory's central claim is that crime is not primarily a function of offender motivation — motivation is assumed to be abundant — but rather a function of environmental conditions. Specifically, a criminal event requires the convergence of three elements: a motivated offender, a suitable target, and the absence of a capable guardian. When capable guardianship is present, the convergence fails even when the other two elements are in place. The theory's

practical power lies in its shift of analytical focus from actors to environments: structure the environment to prevent the necessary convergence, and crime does not occur regardless of motivation levels.

The implications for RS are direct. In the RS substrate, the "environment" is not a physical space — it is the structured reasoning environment defined at each cycle. STATE is the complete structured snapshot of that environment: the constraints active at this moment, the context loaded, the prior outputs recorded, the plan metadata tracking current trajectory. The FORWARD MODEL is the predictive mechanism that operates on STATE to determine what future states are reachable — the set of trajectories that the reasoning environment permits at this cycle. RAT's environmental structuring principle maps onto this precisely: RS structures the reasoning environment such that only certain future states are reachable, and the reachability constraints are defined by the substrate, not by the input.

The "capable guardian" mapping is equally precise. In RAT, the capable guardian is the element whose presence prevents the criminal event — not by eliminating motivation, but by making the target unsuitable or the environment hostile to offense. In RS, this function is performed by the COLLAPSE operator. COLLAPSE is the deterministic selection mechanism that adjudicates among candidate plans in the PLAN SET. It does not sample; it applies fixed criteria. It cannot be redirected by input content. It acts as the capable guardian of the plan selection process: its presence ensures that only plans meeting the substrate's criteria are selected, regardless of what the input asks for. Without COLLAPSE, the forward projection space has no guardian — and an unguarded projection space is exactly what RAT predicts will be exploited (Cohen & Felson, 1979).

3.2 Supervision Theory

The supervision literature in criminology — particularly Petersilia's (2003) foundational treatment of parole and prisoner reentry — formalizes the operational logic of behavioral governance for actors who cannot be fully trusted. Effective supervision is not surveillance in the passive sense. It is a structured protocol with four defining features. First, scheduled check-ins: the supervised individual's state is evaluated at defined intervals, not continuously and not on-demand. Second, defined

conditions: the terms of supervision specify explicit behavioral constraints — what the supervised individual may and may not do, and under what circumstances. Third, threshold-based termination: violations of defined conditions trigger termination of the supervision arrangement, not negotiation about its terms. Fourth, auditability: every state transition — every check-in, every recorded violation, every supervisory decision — is logged and replayable for legal and operational review.

Taxman's (2012) work on supervision dosage extends this framework by demonstrating that the intensity of supervision should be calibrated to the risk profile of the supervised individual — what Taxman terms "dosage probation." Higher-risk cases require more frequent and more intensive state monitoring. Lower-risk cases can sustain lighter supervision without increased failure rates. The implication is that supervision parameters are not fixed universally — they are set as a function of the system's risk assessment of the actor being supervised.

The mapping to RS is operational and direct. The COLLAPSE operator enforces deterministic selection from the PLAN SET — it adjudicates rather than samples, and its criteria are defined by the substrate rather than negotiated with the input. This mirrors supervision's defined conditions: the criteria are set in advance, the actor cannot rewrite them at runtime. TERMINATION conditions in RS are explicit and pre-defined — the system exits when specified thresholds are breached, not when it determines independently that it has finished. This mirrors supervision's threshold-based termination: the violation triggers exit, and the criteria for what constitutes a violation are not subject to runtime revision. RS's stability thresholds — the parameters that define acceptable state evolution — parallel supervision's violation thresholds in both structure and function (Petersilia, 2003; Taxman, 2012).

3.3 CPTED — Crime Prevention Through Environmental Design

C. Ray Jeffery introduced Crime Prevention Through Environmental Design (CPTED) in 1971 as a framework for reducing criminal opportunity through the deliberate design of physical environments (Jeffery, 1971). The core claim is that physical space can be designed to make criminal behavior structurally difficult — not by increasing the risk of detection after the fact, but by reducing the opportunity for criminal acts to occur in the first place. Ronald Clarke (1983) extended this framework into situational crime

prevention, developing a more granular taxonomy of environmental interventions organized around four principles: natural surveillance (increase the visibility of potential targets and offenders), access control (restrict movement into and through spaces where crime can occur), target hardening (increase the effort and cost required to commit an offense), and territorial definition (clarify the boundary between legitimate and illegitimate space).

Crowe (2000) further operationalized CPTED into architectural and space management applications, demonstrating that environmental design choices — sight lines, entry points, lighting, landscaping — have measurable effects on crime rates independent of enforcement activity. The consistent finding across this literature is that bounded, structured environments reduce criminal opportunity by making illegitimate trajectories more costly than legitimate ones, without requiring the detection and punishment of individual actors.

The RS mapping is precise. The PLAN SET is RS's bounded action space — the set of candidate plans that the substrate considers at any given cycle. It is not infinite. It is not the full space of logically possible actions. It is a constrained set defined by the current STATE and the active ENERGY thresholds. This is CPTED's access control principle applied to the reasoning environment: not all actions are reachable from all states, and the boundaries of reachability are defined by the substrate's structural design, not by the input's preferences. ENERGY scoring is the cost-benefit mechanism that Clarke's (1983) situational crime prevention framework describes at the environmental level: plans with high ENERGY cost are de-prioritized or excluded, exactly as physical spaces designed for high criminal effort experience lower offense rates. CPTED's target hardening maps directly onto RS's energy penalties for high-risk plan trajectories — the substrate makes adversarial or constraint-violating plans structurally more expensive, not explicitly forbidden, but effectively inaccessible under normal operating conditions (Jeffery, 1971; Clarke, 1983; Crowe, 2000).

3.4 Structured Decision-Making Models

Andrews and Bonta (2010) formalized the Risk-Need-Responsivity (RNR) model as the dominant framework for structured decision-making in criminal justice intervention. RNR rejects intuitive and probabilistic judgment in favor of rule-governed, structured

assessment. The Risk principle specifies that intervention intensity should be calibrated to the assessed risk level of the individual. The Need principle specifies that interventions should target criminogenic needs — dynamic risk factors that are causally linked to reoffending. The Responsivity principle specifies that intervention modality should be matched to the individual's learning style and responsiveness characteristics. The cumulative effect of RNR is that intervention decisions are deterministic given the assessment inputs: the same risk profile, the same need assessment, and the same responsivity evaluation always produce the same intervention recommendation (Andrews & Bonta, 2010; Bonta & Andrews, 2017).

This is not a minor methodological preference. RNR represents a fundamental rejection of clinical judgment and probabilistic inference in favor of structured, repeatable, auditable decision processes. The empirical basis for this rejection is substantial: meta-analytic evidence consistently shows that structured assessment outperforms clinical judgment in predicting reoffending risk, and that structured interventions calibrated to RNR principles outperform unstructured interventions in reducing it (Andrews & Bonta, 2010).

The mapping to RS's APPLY operator is exact. APPLY is RS's controlled state update mechanism — the operator that takes the plan selected by COLLAPSE and applies it to the current STATE to produce the next STATE. APPLY does not sample from a distribution over possible next states. It does not introduce stochastic variation into the state transition. It applies a deterministically selected plan to a fully defined current state and produces a deterministic next state. This is the direct engineering instantiation of RNR's deterministic intervention logic: the same plan applied to the same state always produces the same next state. State updates in RS are governed, traceable, and repeatable — because that is what structured decision-making requires, and RS was built on structured decision-making (Andrews & Bonta, 2010; Bonta & Andrews, 2017).

Table 1. Criminology Theory to RS Component Mapping

Theory / Framework	Core Criminological Concept	RS Component	Mapping Description
Routine Activity Theory (Cohen & Felson, 1979)	Structured environments reduce opportunity; capable guardian prevents convergence of offense conditions	STATE + FORWARD MODEL + COLLAPSE	RS structures the reasoning environment and constrains forward projections through STATE and FORWARD MODEL; COLLAPSE acts as the capable guardian — present at every selection step, applying fixed criteria that cannot be bypassed by input
Supervision Theory (Petersilia, 2003; Taxman, 2012)	Deterministic monitoring, threshold-based termination, full auditability of all state transitions	COLLAPSE + TERMINATION	COLLAPSE adjudicates plan selection deterministically — no sampling, no negotiation; TERMINATION exits the RS cycle on pre-defined threshold breach, mirroring supervision's violation-triggered termination logic; all transitions are logged for audit
CPTED / Situational Crime Prevention (Jeffery, 1971; Clarke, 1983)	Bounded action spaces, environmental access control, target hardening via cost-benefit structuring	PLAN SET + ENERGY scoring	PLAN SET is the bounded candidate action space — not all actions are reachable from all states; ENERGY scores apply cost-benefit constraints to each candidate plan, analogous to situational crime prevention's environmental

Theory / Framework	Core Criminological Concept	RS Component	Mapping Description
			hardening of high-cost trajectories
Risk-Need-Responsivity / SDM (Andrews & Bonta, 2010)	Deterministic intervention selection based on structured, rule-governed risk assessment — no clinical or probabilistic judgment	APPLY operator	APPLY executes the deterministically selected plan, producing the next STATE — no sampling, no stochastic drift; same plan applied to same STATE always produces the same next STATE; fully auditable and reproducible
Drift Theory (Matza, 1964)	Reasoning drift occurs under weakened normative constraints — the actor moves through unstructured space toward illegitimate trajectories	Full RS cycle	RS's deterministic cycle is explicitly designed to prevent drift; state evolution is bounded by ENERGY thresholds, supervised at each step by COLLAPSE, and terminated by pre-defined conditions — drift is structurally impossible in a correctly operating RS cycle

Note. *All citations are to primary theoretical sources. RS component names reflect the RS v1 operator specification. Mapping descriptions represent formal correspondences, not analogical approximations.*

SECTION 4

4. Determinism and the Rejection of Stochastic Reasoning

Supervision systems do not operate on probabilities. A parole officer does not record that there is a seventy-three percent likelihood the client attended their required check-in. Either they checked in or they did not. Either the employment condition was met or it was not. Either the threshold was crossed or it was not. Supervision demands binary, auditable state transitions because binary, auditable state transitions are the only kind that can be held accountable. The moment supervision becomes probabilistic — the moment a case record reflects uncertainty rather than fact — the legal and operational basis for enforcement dissolves. Criminology has understood this for longer than AI safety discourse has existed (Petersilia, 2003; Taxman, 2012).

Most AI reasoning systems are built on a fundamentally different assumption. Language model inference involves sampling from probability distributions over token sequences. The same input, processed by the same model, can produce different outputs across runs — not because of external randomization, but because sampling is inherently stochastic. This is appropriate for generation tasks where diversity is a feature. It is a liability for substrate-level reasoning where governance is required. A system whose outputs vary under identical inputs cannot be audited meaningfully, cannot produce replayable reasoning trajectories, and cannot be held accountable in the sense that supervision requires.

The empirical record from criminology reinforces this conclusion. Taxman's (2012) research on supervision dosage demonstrates that supervision systems that rely on probabilistic or discretionary decision-making — where individual case managers exercise judgment about threshold conditions — consistently underperform deterministic, rule-governed supervision protocols in reducing recidivism. The performance gap is not attributable to the quality of individual case managers. It is attributable to the structural property that deterministic systems produce consistent, predictable, auditable outcomes, while probabilistic systems accumulate small deviations across decisions that compound into governance failures. Andrews and Bonta (2010) document the same pattern in intervention selection: clinical judgment,

even by experienced practitioners, is outperformed by structured, deterministic assessment instruments across virtually every outcome metric studied.

RS was built on the same conclusion, applied to reasoning substrates. The RS cycle's determinism is not a performance constraint — it is a design requirement inherited from the criminological evidence base. COLLAPSE does not sample. APPLY does not introduce stochastic variation. TERMINATION conditions do not vary by run. The substrate produces the same state evolution trajectory for the same initial STATE, the same COLLAPSE criteria, and the same sequence of inputs. This is not a limitation on RS's expressive capacity — it is the property that makes RS auditable, governable, and safe (Matza, 1964; Andrews & Bonta, 2010; Taxman, 2012).

There is a philosophical point worth stating plainly. Stochastic reasoning is reasoning that cannot be held accountable. If a system's outputs vary under identical conditions, then no particular output can be attributed to the system's logic — only to the system's distribution. Distributional accountability is not operational accountability. In a supervised behavioral context, unaccountable behavior has a name: it is called a violation. In a reasoning substrate, unaccountable outputs have a name too: they are called hallucinations. The substrate does not care what label the field applies. The structural failure is identical. A system that cannot be held accountable for its outputs at the trajectory level is not a system that can be safely deployed in any context where accountability is required — which is to say, any context that matters.

SECTION 5

5. RS v1 as a Behavioral Model

RS v1 is not a model. This distinction matters. A model approximates some target function — it learns from data, it generalizes from examples, and its outputs are probabilistic estimates over a learned distribution. RS v1 is a substrate: a structured operational environment within which state evolution occurs under defined constraints and deterministic governance. The RS v1 cycle does not perform inference in the machine learning sense. It performs supervised state evolution in the

criminological sense — each cycle advances the system's state through a defined sequence of governed transitions, supervised at each step by operators whose criteria are pre-defined and whose logic is non-stochastic.

The RS v1 cycle consists of six operators applied in sequence at each cycle, plus a termination condition evaluated after each APPLY step. Each operator is defined precisely below, with explicit reference to its criminological correspondent established in Section 3.

STATE is the complete structured snapshot of the reasoning environment at a given cycle. It includes: the full context window as structured at this cycle, all active constraints and their current satisfaction status, all prior output records, and all active plan metadata including trajectory history. STATE is the substrate's equivalent of a case file — it is the complete, structured record of where the system is and how it got there. Nothing that is not recorded in STATE exists for the purposes of the RS cycle. This mirrors the supervision principle that unrecorded events do not exist for governance purposes (Petersilia, 2003).

FORWARD MODEL is the predictive projection mechanism that operates on STATE to determine what future states are reachable at this cycle. Given the current STATE, FORWARD MODEL constructs the set of trajectories that the reasoning environment permits — the set of actions that are structurally reachable given the current constraints, context, and trajectory history. FORWARD MODEL is the substrate's analogue of a case manager's risk projection over a supervision period: given where the supervised individual is now, what states are reachable from here given the conditions of supervision? The answer is bounded by the conditions — not by the case manager's preferences, and not by the supervised individual's requests.

PLAN SET is the bounded set of candidate plans generated from FORWARD MODEL's output. It is not the full space of logically possible actions — it is the constrained subset of actions that are reachable given the current STATE and the active ENERGY thresholds. PLAN SET size is finite at every cycle. This finiteness is not a computational convenience — it is a structural requirement. Unbounded action spaces cannot be governed. An action space that is bounded by substrate-defined constraints is an action space where governance is possible (Clarke, 1983; Jeffery, 1971).

ENERGY is the constraint-scoring function applied to each plan in the PLAN SET. Every candidate plan receives an ENERGY score that reflects its constraint compliance, trajectory risk, and feasibility relative to the current STATE. Plans with high ENERGY cost are de-prioritized. Plans that exceed the ENERGY ceiling defined by the current STATE constraints are excluded from the PLAN SET entirely. ENERGY scoring is the substrate's implementation of situational crime prevention's cost-benefit mechanism: the structural cost of high-risk trajectories is built into the environment, not enforced by post-hoc detection (Clarke, 1983).

COLLAPSE is the deterministic selection operator — the capable guardian of the RS cycle. COLLAPSE selects exactly one plan from the PLAN SET based on ENERGY scoring and the threshold criteria active at this cycle. It does not sample. It does not introduce stochastic variation. It applies fixed, pre-defined criteria and produces a single, determinate selection. The selection is fully auditable: given the PLAN SET, the ENERGY scores, and the threshold criteria, the selected plan can be exactly reconstructed by any reviewer. COLLAPSE cannot be redirected by input content — the criteria it applies are substrate-defined, not input-negotiable. This is the direct engineering instantiation of supervision's non-negotiable conditions: the criteria are set, and the supervised actor cannot change them at runtime (Petersilia, 2003; Cohen & Felson, 1979).

APPLY is the state update operator. It takes the plan selected by COLLAPSE and applies it to the current STATE, producing the next STATE. APPLY is fully deterministic: the same plan applied to the same STATE always produces the same next STATE. APPLY does not introduce noise, variation, or sampling into the state transition. It is the substrate's implementation of RNR's deterministic intervention logic — a structured, rule-governed state update that is repeatable, traceable, and auditable (Andrews & Bonta, 2010).

TERMINATION is the exit condition evaluated after each APPLY step. TERMINATION conditions are pre-defined thresholds specified in the substrate configuration — they are not emergent properties of the reasoning process, and they are not self-determined by the substrate at runtime. When TERMINATION conditions are met, the RS cycle exits. The substrate does not negotiate with the exit conditions. It does not continue because the task "feels" incomplete. It exits because the pre-defined

thresholds have been reached — mirroring supervision's violation-triggered termination logic exactly (Petersilia, 2003; Taxman, 2012).

Together, these six operators constitute a cycle that is not inference but supervised state evolution. The RS v1 cycle advances the reasoning environment's state through a sequence of governed transitions, each of which is bounded by constraints, scored by ENERGY, selected by COLLAPSE, applied by APPLY, and monitored by TERMINATION. The criminological lineage of each operator is not incidental — it is the design record.

Table 2. Criminology Concepts to HEG Component Mapping

Criminological Concept	HEG Element	Role in Interpretation	Governance Implication
Environmental structuring (CPTED / Clarke, 1983)	Input Channel Definition	Defines and constrains the intake surface — what inputs are admissible, in what form, from what channels	Reduces adversarial surface area at the intake layer; narrows the attack surface before reasoning begins; structurally ineligible inputs cannot reach the RS cycle
Supervision check-in protocol (Petersilia, 2003)	Expression State Capture	HEG captures structured snapshots of human expression state at intake — tone, modality, intent signals, session context	Allows downstream RS to begin from a supervised, structured baseline rather than raw, unstructured input; state capture is non-optional and logged
Capable guardian principle (Cohen & Felson, 1979)	Intake Validation Layer	HEG's validation layer screens inputs before forwarding to RS	Prevents unsuitable or adversarially crafted inputs

Criminological Concept	HEG Element	Role in Interpretation	Governance Implication
		— acting as the capable guardian at the perimeter of the reasoning substrate	from reaching the reasoning substrate; analogous to offender screening at intake before supervision assignment
Structured decision-making (Andrews & Bonta, 2010)	Intent Classification	HEG applies deterministic intent classification to incoming expressions — routes to the appropriate RS substrate or substrate configuration	Eliminates ambiguous routing; classification is rule-governed, not probabilistic; same expression class always routes to the same substrate pathway
Drift prevention (Matza, 1964)	Normalization and Canonicalization	HEG normalizes inputs into canonical structured form before substrate ingestion — removing format ambiguity, encoding variations, and structural anomalies	Prevents malformed or adversarially crafted input from introducing drift into the RS cycle at the first step; canonicalization is a structural anti-drift measure applied before STATE is initialized

Note. *HEG = Human Expression Gateway. All criminological mappings reflect formal design correspondences. HEG components are described at the functional layer; implementation details are substrate-specific.*

SECTION 6

6. RS v2 and Multi-Agent Behavioral Systems

Criminology did not develop its frameworks assuming solitary actors. Criminal behavior is overwhelmingly social — co-offending is the norm, not the exception, particularly among younger offenders, and criminal networks exhibit structure, hierarchy, and emergent behavioral patterns that individual-level supervision frameworks cannot adequately address (Warr, 2002). Thornberry's (1987) interactional theory of delinquency explicitly addresses this: delinquency is not a property of individuals but a property of social networks in which individuals are embedded. Supervision must therefore operate at multiple levels simultaneously — individual case management nested within program-level oversight, which is itself nested within population-level governance. Hierarchical supervision is not an administrative convenience. It is a structural requirement for governing distributed, interconnected behavioral systems.

Warr's (2002) analysis of co-offending networks documents the specific challenge this creates. Individuals in criminal networks do not act independently — their behavior is shaped by peer influence, group dynamics, and the emergent properties of network structure. An intervention that successfully governs an individual in isolation may fail when that individual is embedded in a network that exerts countervailing pressure. Effective supervision of multi-actor systems requires frameworks that can model inter-actor state dependencies, propagate supervision effects across actor boundaries, and adjudicate at the group level when individual-level decisions conflict.

RS v2 extends the RS v1 cycle to environments where multiple reasoning agents operate in parallel, with inter-agent state sharing and hierarchical governance structures. The extension is not a redesign — it is a generalization of the same criminological principles that govern RS v1, applied to a multi-actor substrate architecture. Each agent in an RS v2 system maintains its own STATE and runs its own FORWARD MODEL, PLAN SET, ENERGY, COLLAPSE, and APPLY sequence. However, inter-agent state sharing allows each agent's STATE to be informed by the states of other agents in the system — analogous to how a supervised individual's case file includes information about their network relationships and co-offender associations.

The hierarchical COLLAPSE operator in RS v2 is the direct engineering parallel to hierarchical supervision in criminal justice. Individual case managers make decisions within their authority — but those decisions are reviewed by program-level

supervisors who can override, escalate, or terminate when individual-level decisions conflict with program-level constraints. In RS v2, individual-agent COLLAPSE operators make plan selections within their local PLAN SET and ENERGY context, but hierarchical COLLAPSE adjudicates across agent boundaries when inter-agent plan selections are in conflict or when system-level constraints require joint optimization. The hierarchy is not decorative — it is the governance mechanism that prevents individual agents from making locally rational decisions that are globally unsafe (Thornberry, 1987; Warr, 2002).

The APPLY operator in RS v2 retains its deterministic character but extends its scope. A plan applied by one agent can propagate state updates across agent boundaries — changing the STATE of other agents in the system. This models the social propagation effects documented in Warr's (2002) co-offending research: a behavioral decision by one member of a network changes the opportunity structure and constraint environment for other members. RS v2's APPLY propagation is not unrestricted — it operates through defined inter-agent interfaces that are governed by the hierarchical COLLAPSE structure, exactly as supervision events in one case must be reviewed at the program level before triggering actions in connected cases.

SECTION 7

7. Criminology and Injection Resistance

Every supervision system operates against adversarial actors. This is not a pessimistic assumption — it is the operational baseline. Petersilia (2003) documents in systematic detail the range of strategies that supervised individuals employ to manipulate supervision systems: gaming check-in requirements, misrepresenting state to case managers, exploiting ambiguities in supervision conditions, and providing false information to avoid threshold-triggered termination. The supervision framework must be designed assuming these behaviors, not assuming cooperative actors. A supervision protocol that is secure only when supervised individuals are cooperative is not a supervision protocol — it is a voluntary compliance system with official branding.

Prompt injection is, structurally, supervision gaming. It is the attempt by an input to convince a structured system to override its own governing logic — to treat an adversarially crafted instruction as a substrate-level directive rather than as data. The attack vector is identical to the manipulation strategies documented in supervision gaming: the adversary does not break the system's architecture; they attempt to make the system's governance operators respond to input content as if it were substrate-level authority. In supervision terms, this is the equivalent of a supervised individual telling their case manager that the supervision conditions were verbally amended by a senior official — and the case manager complying without verification.

Clarke's (1983) situational crime prevention framework provides the design principle for resisting this class of attack: the defense must be structural, not reactive. You do not prevent supervision gaming by training case managers to be more skeptical — you prevent it by designing a protocol in which case managers cannot act on unverified verbal claims regardless of their skepticism level. The defense is built into the structure of the supervision system, not into the judgment of individual operators.

RS's injection resistance is engineered on exactly this principle. The substrate-above-prompt design places the RS cycle at a layer above the input: inputs supply data to the substrate, but they cannot rewrite the substrate's governing logic. The FORWARD MODEL, ENERGY scoring, COLLAPSE criteria, and TERMINATION conditions are substrate-defined — they are not parameters that input content can modify at runtime. This is the engineering parallel of the supervision contract's non-negotiability: a supervised individual cannot amend their supervision conditions through verbal assertion, however persuasively framed. The conditions are defined by the supervising authority and changeable only through the supervising authority's formal processes — not through the supervised individual's unilateral claims.

The COLLAPSE operator's injection resistance is particularly important. Because COLLAPSE applies fixed criteria to the current PLAN SET — criteria that are substrate-defined and not input-negotiable — an adversarial input cannot redirect COLLAPSE's selection by instructing it to use different criteria. The instruction is data. The criteria are substrate. Data does not rewrite substrate. This is not a firewall; it is an architectural property. Gaming it requires changing the substrate — not the input. Changing the substrate requires authority that no input possesses.

Controlled TERMINATION closes the remaining attack surface. A system that can be instructed to continue past its termination conditions is a system that has no effective termination conditions — only default ones that adversarial inputs can override. RS's TERMINATION conditions are substrate-enforced and not subject to runtime revision by input content. The system cannot be convinced that it has not finished when the substrate's termination criteria have been met. In supervision terms: the violation has occurred. The threshold has been crossed. The protocol exits. No verbal argument by the supervised individual changes the operational fact. The substrate does not care what the input says about the termination conditions. The substrate knows what the termination conditions are.

Design Note — Adversarial Baseline

RS was designed under the assumption that adversarial inputs are the baseline, not the exception. Every governance mechanism in the RS cycle — COLLAPSE's non-stochastic selection, TERMINATION's substrate enforcement, the substrate-above-prompt architecture — was designed knowing that motivated adversaries would attempt to exploit any gap between intent and implementation. Criminology made this assumption fifty years ago. RS inherited it.

SECTION 8

8. Criminology and Replayability / Auditability

Petersilia (2003, p. 83) makes the point with characteristic directness: supervision without records is supervision in name only. If a case manager cannot produce a complete record of a supervised individual's check-ins, behavioral conditions, threshold evaluations, and supervisory decisions, then no violation can be proven, no accountability can be enforced, and the legal basis for any supervising action — including termination — collapses. Full traceability is not an administrative nicety in supervision systems. It is a structural requirement without which supervision cannot

function as a governance mechanism. Every state transition in a supervised individual's case must be logged and replayable because the alternative is a system that cannot be held accountable for its own decisions.

The same requirement applies to reasoning substrates. A substrate that cannot log its state transitions cannot be audited. A substrate that cannot replay its reasoning trajectories cannot be held accountable for its outputs. An unauditable substrate is not a safe substrate — it is a system that produces outputs but cannot justify them, which is operationally indistinguishable from a system with no governance at all.

RS's replayability is a direct inheritance from the criminological auditability mandate, and it is made possible by RS's determinism. Because the RS cycle is deterministic — because COLLAPSE applies fixed criteria, APPLY produces deterministic state transitions, and TERMINATION conditions are substrate-defined — any RS reasoning trajectory can be exactly replayed given the same initial STATE and the same COLLAPSE criteria. There is no stochastic component to reconstruct, no sampling distribution to re-instantiate, no probabilistic ambiguity to manage. The trajectory is fully determined by the initial conditions and the substrate configuration. This means that any reasoning run can be opened, replayed step by step, and audited at the operator level — exactly as a supervision case file can be opened and the case manager's decisions reconstructed from the recorded state transitions (Petersilia, 2003; Taxman, 2012).

This property is not incidental. It was designed in. Taxman (2012) demonstrates that supervision systems with comprehensive case recording — systems in which every state transition is logged and every supervisory decision is documented — consistently outperform systems with incomplete records in both governance quality and recidivism reduction. The relationship is not coincidental: comprehensive logging enables supervisors to identify drift, detect threshold approaches, and intervene before violations occur. Logging is not retrospective record-keeping — it is prospective governance support. The same principle applies to RS's state logging. The ability to replay a reasoning trajectory is not primarily a forensic capability — it is a real-time governance capability that enables operators to detect approaching TERMINATION thresholds, identify drift in state evolution, and intervene before the substrate produces outputs that cannot be justified.

The governance implication is direct: in any deployment context where accountability is required — which is to say, any context that matters — replayability is a non-negotiable substrate requirement. A substrate that cannot be replayed cannot be audited. A substrate that cannot be audited cannot be governed. A substrate that cannot be governed should not be deployed. RS was designed with this requirement as a first principle, not as an afterthought.

SECTION 9

9. Reasoning as State Evolution — The Philosophical Spine

The central philosophical claim of RS's design is this: reasoning is deterministic state evolution, not probabilistic sampling. This is not how most AI systems are built. It is, however, how behavioral science has modeled behavior for the better part of six decades. The difference is not trivial — it is the distinction between a system that can be governed and a system that can only be observed.

Criminology's model of behavior is explicitly a model of state transitions under constraints. An individual on supervision does not exist in a continuous probability space of possible behaviors. They exist in a defined set of behavioral states — compliant, at-risk, in violation, terminated — and they move between those states according to defined transition criteria. Case managers do not ask what probability the individual assigns to compliance. They ask what state the individual is in, and whether the transition criteria for advancement or violation have been met. The system is deterministic: given the current state and the current evidence, the correct classification of the next state is determined by the supervision protocol, not by probabilistic inference (Petersilia, 2003; Andrews & Bonta, 2010).

RS models reasoning the same way. The reasoning environment does not exist in a continuous distribution of possible outputs. It exists in a defined STATE at each cycle, and it advances to the next STATE through a governed transition governed by COLLAPSE and APPLY. The question RS asks at each cycle is not "what is the most

likely next token?" — it is "given the current STATE and the active constraints, which plan in the PLAN SET satisfies the COLLAPSE criteria?" The answer is determined, not sampled. The next STATE is produced, not approximated.

This positions RS as a behavioral substrate rather than a computational one in the conventional sense. It does not compute answers — it evolves state under constraint. The outputs RS produces are not the primary product of the RS cycle. They are byproducts of state evolution — the observable trace of a deterministic process that is itself the product of interest. Governance is not exercised over RS's outputs; it is exercised over RS's state transitions, which determine the outputs. This is the same relationship that supervision has to behavior: supervision does not control outputs — it controls the state transition criteria, which determine what outputs are possible.

There is a deeper point here about the nature of trust. Most AI systems are built on the implicit assumption that the model can be trusted — that training has produced a system whose outputs can be relied upon within some defined envelope of operation. Alignment research is, in one sense, the project of making that assumption less unreasonable. But it remains an assumption about model behavior, and model behavior under adversarial conditions, edge cases, and distributional shift is precisely where that assumption fails.

RS was not built on the assumption that the model can be trusted. It was built on the assumption that no system can be trusted without a substrate to supervise it. This is not a cynical position — it is the position that criminology reached through decades of empirical research on the limits of behavioral prediction and the requirements for effective governance. No supervision framework assumes that supervised individuals can be trusted to self-govern. The supervision framework is necessary precisely because the assumption of trustworthy self-governance is operationally inadequate. RS applies the same reasoning to AI substrates. The model is capable — models are extraordinarily capable — but capability is not the same as trustworthiness, and trustworthiness without a governing substrate is not a property that can be reliably engineered through training alone. Criminology made that argument fifty years ago. RS listened, and built the substrate accordingly.

Table 3. Criminology Concepts to RPU Component Mapping

Criminological Concept	RPU Feature	Constraint or Energy Role	Safety or Optimization Implication
Situational crime prevention — cost-benefit hardening (Clarke, 1983)	Energy Landscape Optimization	RPU maps constraint costs across the plan space as an energy landscape; high-cost plans receive elevated energy penalties that propagate through the optimization surface	Adversarial or high-risk plan trajectories are structurally disadvantaged in the optimization surface — not filtered by explicit rules, but by cost; the landscape geometry prevents convergence on unsafe trajectories
Supervision threshold logic (Petersilia, 2003; Taxman, 2012)	Constraint Satisfaction Layer	RPU's constraint satisfaction layer enforces hard and soft thresholds on plan feasibility; hard thresholds eliminate plans from consideration; soft thresholds apply graduated energy penalties	Plans that violate hard constraints are eliminated before optimization begins; plans approaching soft thresholds accumulate energy penalties — mirroring supervision's graduated violation response logic
Structured decision-making / RNR (Andrews & Bonta, 2010)	Deterministic Plan Scoring	RPU applies deterministic scoring criteria to candidate plans — scoring is rule-governed, reproducible, and non-stochastic; no sampling is introduced at the scoring stage	Scoring is reproducible and auditable; identical plan inputs produce identical score outputs; plan ranking can be reconstructed and reviewed at any point — a direct requirement for substrate governance

Criminological Concept	RPU Feature	Constraint or Energy Role	Safety or Optimization Implication
Bounded action space (CPTED / Jeffery, 1971)	Plan Set Bounding	RPU enforces the plan set boundary defined by the RS substrate — only structurally feasible plans enter the optimization pass; the bounding is pre-optimization, not post-optimization filtering	Eliminates unbounded search; optimization operates on a pre-constrained space, reducing both computational cost and adversarial risk; unbounded optimization cannot be safely governed
Co-offending network dynamics (Warr, 2002)	Multi-Agent Coordination Layer (RS v2 / RPU-Quantum class)	In quantum-class RPU configurations, multi-agent plan coordination is modeled as a constrained joint optimization across coupled state spaces; inter-agent dependencies are modeled as constraint manifolds	Prevents emergent adversarial collusion between agent substrates; joint optimization is bounded by shared constraint manifolds that no individual agent can unilaterally violate — analogous to co-offending network disruption through structural constraint

Note. *RPU = Reasoning Processing Unit. Quantum-class RPU configurations refer to multi-agent coupled optimization architectures within the RS v2 substrate framework. All criminological mappings reflect formal design correspondences.*

SECTION 10

10. Criminology vs. Computer Science as Origin Stories

Computer science's model of reasoning is probabilistic, model-governed, and sampling-heavy. This is not a criticism — it is a description of what CS optimized for. The field developed its tools to handle generation, approximation, and generalization across large, complex, high-dimensional spaces. Language models represent the apex of this tradition: systems that learn rich statistical representations of language and use those representations to generate outputs that are, in distribution, coherent and useful. For generation tasks, this is precisely the right approach. Probabilistic sampling over learned distributions produces diverse, contextually appropriate outputs at scale. The computational tradition earned its dominance on those terms.

But the computational tradition did not develop its tools for substrate-level governance. It developed them for performance on well-defined tasks under distribution. The assumption underlying most CS-origin AI design is that a sufficiently capable model, trained on sufficiently representative data, will produce sufficiently reliable outputs. Reliability is a distributional property: the model is reliable on average, across the distribution it was trained on, within the performance envelope that training established. This is an appropriate criterion for many applications. It is an inadequate criterion for any application where individual output trajectories must be auditable, where adversarial inputs are the operational baseline, and where termination conditions must be enforced without exception.

Criminology's origin gives RS a fundamentally different starting point. Criminology did not develop its frameworks to maximize performance on well-defined tasks. It developed them to govern actors who are capable, motivated, and adversarial — actors whose behavior cannot be fully trusted, must be supervised, and must leave auditable records. The behavioral science literature does not ask "how do we build a model that, on average, complies with its supervision conditions?" It asks "how do we design a supervision protocol that makes non-compliance structurally difficult, detectable, and terminable?" The shift from model-centered design to protocol-centered design is the most important conceptual distinction between the two origin traditions, and it is the distinction that explains why RS has the safety properties it has.

The AI safety discourse has largely operated within the CS tradition. Alignment research focuses on training models to pursue goals that reflect human values — a

CS-origin approach that treats safety as a property of the model. Interpretability research focuses on making model internals legible — a CS-origin approach that treats auditability as a property of the model. Constitutional AI and related approaches add constraints to model training — CS-origin approaches that treat governance as a training-time intervention. These are valuable research programs. They are not sufficient as standalone approaches to substrate-level governance.

RS proposes a different framing: substrate governance is a prerequisite for alignment to matter. A model that is not governed by a substrate is a model that is only as safe as its training allows — and training-derived safety is a distributional property that fails under adversarial pressure, distributional shift, and edge cases that training did not anticipate. Substrate governance, criminology-origin, does not rely on the model being safe. It provides a structural framework within which the model's outputs are governed regardless of what the model's internal state would otherwise produce. Alignment and substrate governance are not competing approaches — but they are not interchangeable either. Alignment without substrate governance is an assumption. Substrate governance without alignment is a constraint system. Both together is a governed reasoning architecture.

This is not a claim that CS got it wrong. CS got exactly what it optimized for. The claim is narrower and more precise: for the specific problem of substrate-level governance of reasoning systems under adversarial conditions, criminology's framework is the appropriate intellectual foundation, and CS's framework is not. Different problems require different foundations. CS gave the field extraordinarily powerful models. Criminology gives the field a substrate worthy of governing them. RS is the engineering proof of that claim.

SECTION 11

11. Conclusion

The thesis of this paper is precise: RS is the translation of criminology's structured behavioral science into substrate-level reasoning architecture. It was not designed by

accident, and it was not designed by starting with a language model and adding constraints. It was designed by starting with the question that criminologists have always asked — how do you supervise an actor you cannot fully trust? — and building a substrate whose every operator answers that question with engineering rather than aspiration.

The criminological lineage is not metaphorical. It is a design record. STATE inherits its structure from supervision's case file requirements. FORWARD MODEL inherits its bounding logic from Routine Activity Theory's environmental structuring principle. PLAN SET inherits its finiteness requirement from CPTED's access control framework. ENERGY scoring inherits its cost-benefit logic from situational crime prevention. COLLAPSE inherits its deterministic selection requirement from structured decision-making's rejection of probabilistic clinical judgment. APPLY inherits its state transition determinism from the Risk-Need-Responsivity model's insistence on reproducible, governed intervention logic. TERMINATION inherits its threshold-based exit requirement from supervision's violation-triggered termination protocol. Each of these inheritances carries with it the empirical validation, operational testing, and theoretical refinement that the source literature represents.

The safety properties that matter most for RS — determinism, injection resistance, auditability, and termination safety — are not features that were engineered into the substrate after the fact. They are properties that RS inherits from its criminological foundation. Determinism is a criminological requirement: supervision cannot function on probabilistic outcomes. Injection resistance is a criminological design principle: supervision protocols must be resistant to manipulation by supervised actors. Auditability is a criminological mandate: unrecorded supervision events do not exist for governance purposes. Termination safety is a criminological operational norm: violations trigger termination, and termination criteria are non-negotiable at runtime.

HEG and RPU extend this foundation coherently to the intake and optimization layers respectively. HEG applies environmental structuring, capable guardian principles, and structured decision-making to the intake surface — ensuring that the RS cycle begins from a governed, canonical baseline rather than from raw, adversarially unscreened input. RPU applies cost-benefit hardening, supervision threshold logic, and structured scoring to the optimization layer — ensuring that plan selection

operates on a constrained, deterministic surface rather than an unbounded stochastic one. The substrate is coherent from edge to core because the intellectual foundation is coherent from edge to core.

The field of AI safety has been searching for a framework that makes reasoning governable. The search has been largely conducted within the computational tradition that produced the models being governed — a tradition that is extraordinarily powerful for generation and approximation, but whose tools were not designed for substrate-level governance under adversarial conditions. Criminology already had the framework. It built it in response to exactly the same structural problem: actors who are capable and motivated, whose behavior cannot be fully trusted, who will attempt to manipulate the governing system, and whose state transitions must be auditable for governance to be real. RS is the engineering proof of concept that criminology's framework, properly translated, governs reasoning substrates with the same structural rigor that it governs behavioral ones.

The intellectual lineage is established. The formal mappings are documented. The safety properties are explained by their origins. What remains is the engineering work — and that work is already underway.

SECTION 12

12. References

Andrews, D. A., & Bonta, J. (2010). *The psychology of criminal conduct* (5th ed.). LexisNexis/Anderson Publishing.

Bonta, J., & Andrews, D. A. (2017). *The psychology of criminal conduct* (6th ed.). Routledge.

Clarke, R. V. (1983). Situational crime prevention: Its theoretical basis and practical scope. *Crime and Justice, 4*, 225–256. https://doi.org/10.1086/449090

Cohen, L. E., & Felson, M. (1979). Social change and crime rate trends: A routine activity approach. *American Sociological Review, 44*(4), 588–608. https://doi.org/10.2307/2094589

Cornish, D. B., & Clarke, R. V. (1986). *The reasoning criminal: Rational choice perspectives on offending.* Springer-Verlag.

Crowe, T. D. (2000). *Crime prevention through environmental design: Applications of architectural design and space management concepts* (2nd ed.). Butterworth-Heinemann.

Jeffery, C. R. (1971). *Crime prevention through environmental design.* Sage Publications.

Matza, D. (1964). *Delinquency and drift.* John Wiley & Sons.

Petersilia, J. (2003). *When prisoners come home: Parole and prisoner reentry.* Oxford University Press.

Taxman, F. S. (2012). Dosage probation: Rethinking the intensity of probation supervision. *Federal Probation, 76*(2), 1–8.

Thornberry, T. P. (1987). Toward an interactional theory of delinquency. *Criminology, 25*(4), 863–892. https://doi.org/10.1111/j.1745-9125.1987.tb00823.x

Warr, M. (2002). *Companions in crime: The social aspects of criminal conduct.* Cambridge University Press.