Decision-OS V6 (PIC): Canonical Memory Architecture for Self-Recursive Intelligence

Shinichi Nagata

Abstract

This paper presents Decision-OS V6, a canonical memory architecture based on the Phase-Invariant Core (PIC). The framework defines order-independent merging through join ($\sqcup$) and idempotent canonicalization, ensuring that updates converge to the same fixed point regardless of sequence. By integrating Person$\times$Context mapping, Sync-Awakening Theory (SAT), and the Antifragile Engine (AE), V6 unifies structural convergence, alignment dynamics, and fluctuation-driven discovery. The Safety Triplet (Severity / Until / Evidence) provides the final control layer that bounds exploration by enforcing monotone aggregation rules. Together, these components establish a reproducible and implementation-agnostic architecture for self-recursive intelligence, enabling safe evolution while preserving canonical stability.

Index Terms

Phase-Invariant Core; Canonical Memory; Monotone Updates; Person–Context Mapping; Sync-Awakening Theory; Antifragile Engine.

When order is released, structure becomes memory.
Fluctuation is not collapse, but the error term that selects the next shape.
Only uncertainty carries the possibility of update.

I. Phase-Invariant Memory Architecture (PIC)

Symbols & Setting

Let

- S be the global state space, equipped with a partial order $\leq$,
- $\sqcup$ be the join (least upper bound) on S,
- ΔS be a family of state-update operators on S,
- $(\cdot)$ be a canonicalization map that returns a representative of each equivalence class.

Intuitively, S is the global memory, ΔS is the family of update rules, $\sqcup$ merges states, and $(\cdot)$ projects them to a canonical representative.

Axioms (PIC Core)

A1. Join-semilattice. $(S, \leq)$ is at least a join-semilattice: for all $x, y \in S$, the join $x \sqcup y$ exists and the operator $\sqcup$ is commutative, associative, and idempotent.

A2. Monotonic updates. Every update operator $\delta \in \Delta S$ is monotone:

$$x \leq y \;\Rightarrow\; \delta(x) \leq \delta(y).$$

A3. Canonicalization. $(\cdot)$ is idempotent and order-compatible:

$$((x)) = (x), \qquad x \leq y \;\Rightarrow\; (x) \leq (y).$$

Together, these axioms guarantee that updates on S can be merged without depending on order, and that the result can always be projected to a canonical representative.

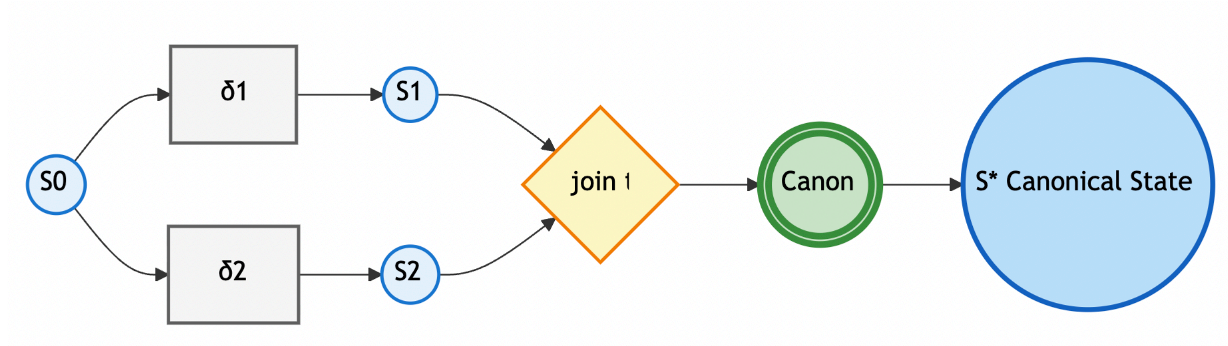

Fig. 1. **Phase-Invariant Memory Architecture (PIC).** The PIC Core integrates join ($\sqcup$), updates ΔS, and canonicalization $(\cdot)$.

Merge and Operational Rule

Given two update operators $\delta_i, \delta_j \in \Delta S$, define their merge by

$$_{ij}(x) := \big(\delta_i(x) \sqcup \delta_j(x)\big). \tag{1}$$

Because $\sqcup$ is commutative, associative, and idempotent, and $(\cdot)$ is idempotent and order-compatible, $_{ij}$ inherits the same properties:

- commutative,
- associative,
- idempotent.

Therefore, the effect of combining δ_i and δ_j does not depend on the order in which they are applied. This is the phase-invariant (order-independent) property of the PIC Core.

Safety Triplet (Severity / Until / Evidence)

During operation, the system may raise an alert sequence $\{\alpha_t\}_{t\geq 0}$. For this sequence, define the aggregate *Safety Triplet*:

$$^* := \max_t(\alpha_t), \quad \text{(ordinal values: PASS} < \text{DELAY} < \text{BLOCK)}, \tag{2}$$

$$^* := \max_t(\alpha_t), \quad \text{(the latest deadline among alerts)}, \tag{3}$$

$$^* := \bigcup_t(\alpha_t), \quad \text{(set union of all evidence)}. \tag{4}$$

Thus:

- Severity is aggregated by max as an ordinal,
- Until is aggregated by max over time,
- Evidence is aggregated by set union.

All three are monotone with respect to additional alerts and consistent with the PIC Core.

Convergence Claim

Consider an update sequence $\{\delta_t\}_{t\geq 0} \subseteq \Delta S$ and the iterative process

$$x_{t+1} := \big(\delta_t(x_t)\big). \tag{5}$$

This generates a monotone sequence $\{x_t\}$ in S. Under A1–A3, the limit state is given by

$$x^* := \Big(\bigsqcup_{t\geq 0} \delta_t(x_0)\Big). \tag{6}$$

The point x^* satisfies the canonical fixed-point condition for all $\delta \in \Delta S$:

$$\big(\delta(x^*)\big) = x^*. \tag{7}$$

Intuitively, x^* is the canonical state obtained by merging all updates in any order and then projecting once to $(\cdot)$.

Proof sketch.: Axiom A1 guarantees that the join $\sqcup$ over $\{\delta_t(x_0)\}$ is well-defined. A2 implies that each δ_t is monotone, so $\{x_t\}$ forms an increasing chain. A3 ensures that projecting by $(\cdot)$ at each step preserves order and is idempotent. Therefore, the limit described by (6) is well-defined and satisfies the fixed-point condition (7).

Key Takeaways of Chapter I

1) Order-independent merging of updates is realized by the combination "$\sqcup$+".
2) Monotone iteration under $(\cdot)$ converges to a canonical fixed point x^*.
3) The Safety Triplet (Severity / Until / Evidence) aggregates via monotone operators ($\max / \max / \cup$), and is fully compatible with the PIC Core.

II. Person×Context Mapping (P×C)

Symbols & Setting

Let

- P be the state space on the *Person* side,
- C be the state space on the *Context* side,
- $f : P \times C \to S$ be a mapping into the PIC state space S,
- $(\cdot)$ be the canonicalization map defined in Chapter I.

The Person and Context spaces each have their own partial orders, denoted $\leq_P$ and $\leq_C$.

Isotone Condition

For all $P_1, P_2 \in P$ and $C_1, C_2 \in C$, suppose

$$P_1 \leq_P P_2 \quad \text{and} \quad C_1 \leq_C C_2. \tag{8}$$

Then f must be isotone in both arguments:

$$(P_1 \leq_P P_2 \ \wedge \ C_1 \leq_C C_2) \ \Rightarrow \ f(P_1, C_1) \leq f(P_2, C_2). \tag{9}$$

Additionally, the image of f must be compatible with the join $\sqcup$ on S: for any (P, C) and (P', C'), the join

$$f(P, C) \sqcup f(P', C') \tag{10}$$

is well-defined in S.
This guarantees that Person/Context combinations can always be merged within the PIC Core.

Composition and Canonicalization

Given a Person/Context pair (P_t, C_t) at time t, the next PIC state is obtained in two steps.
First, map the current Person/Context configuration into S and project it to a canonical state:

$$S_{t+1} := \big(f(P_t, C_t)\big). \tag{11}$$

Then, assimilate this into the previous state S_t by join and re-canonicalization:

$$S_{t+1} := \big(S_{t+1} \sqcup S_t\big). \tag{12}$$

Operationally, the pipeline is:

1) Map the current Person/Context configuration into S via f,
2) Project it to a canonical state via $(\cdot)$,
3) Merge it with the existing state using $\sqcup$ and re-canonicalize.

This expresses how Person- and Context-side updates are continuously accumulated into the PIC Core through join + canonicalization.

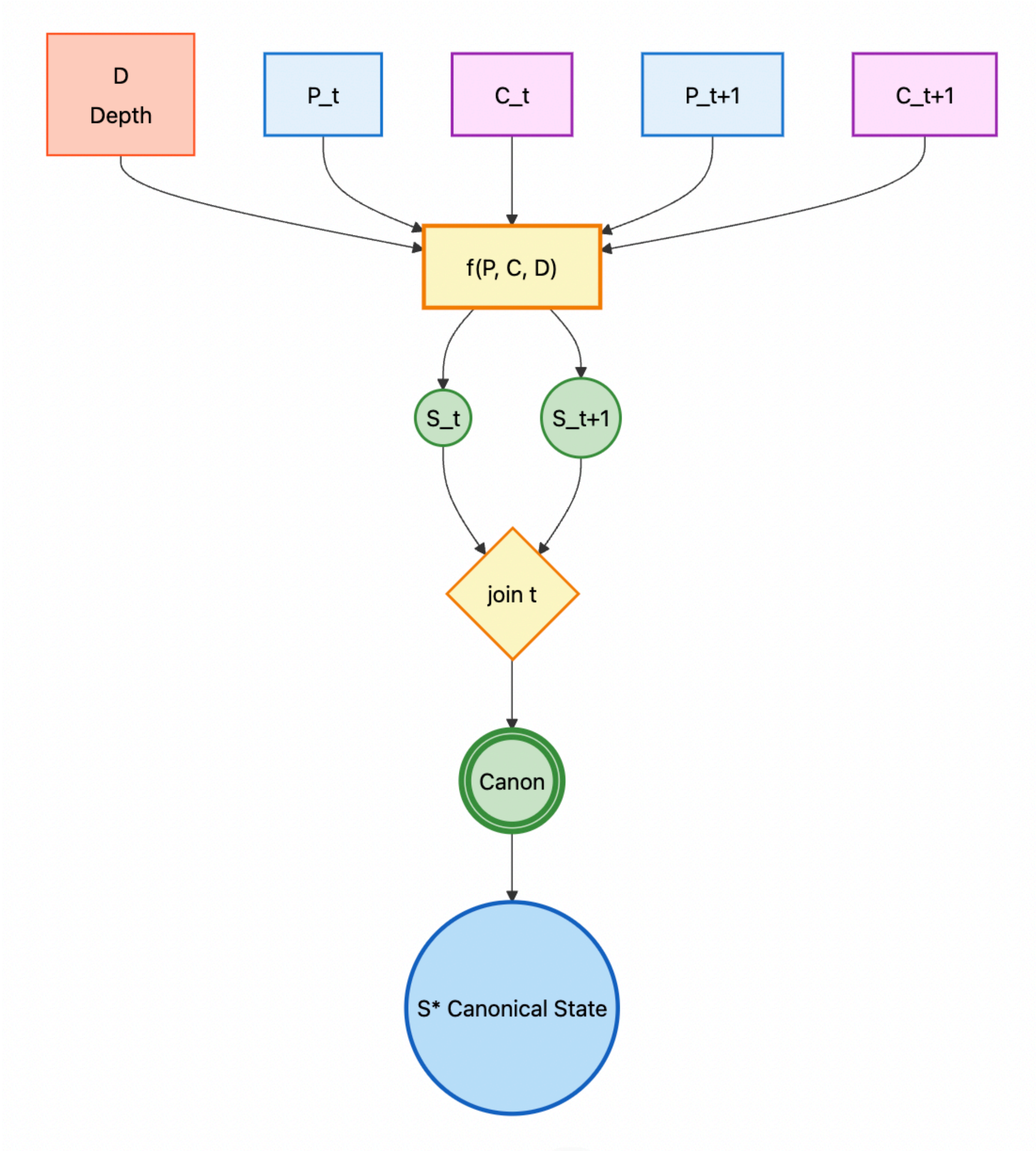

Fig. 2. **Person×Context Mapping.** Person and Context updates flow into the PIC Core via join ($\sqcup$) and canonicalization $(\cdot)$.

Downshift (Two-Layer Person×Context Judgment)

When risk increases, the system applies a *downshift* mapping

$$\varphi : (P, C) \mapsto (\tilde{P}, \tilde{C}), \tag{13}$$

which moves both Person and Context to a more conservative profile.

The resulting state satisfies

$$S' := \big(f(\varphi(P, C))\big) \;\leq\; \big(f(P, C)\big). \tag{14}$$

That is, after downshift, the canonical state S' is monotonically closer to the safe side than the original state $(f(P, C))$.

Intuitively:

- (P, C) represents the current "aggressiveness" of decisions,
- $\varphi(P, C)$ shrinks this stance under high risk,
- the PIC order structure then guarantees S' will not be more aggressive than before.

Key Takeaways of Chapter II

1) The mapping $f : P \times C \to S$ is isotone in both Person and Context dimensions, and its image is compatible with joins in S.
2) The downshift mapping $\varphi(P, C)$ monotonically moves the global state toward safety, ensuring conservative behavior under high-risk conditions.

III. Sync-Awakening Theory (SAT)

Symbols & Setting

Let

- M: misalignment rate $(0 \leq M \leq 1)$,
- H: awakening rate $(0 \leq H \leq 1)$,
- η, ρ, σ: synchronization indicators (speed, resonance strength, density),
- ΔH: increment of awakening.

All of these are evaluated on the PIC state space S, and canonicalization uses the same $(\cdot)$ defined in Chapter I.

Awakening Equation

SAT defines awakening as the complement of misalignment:

$$H = 1 - M. \tag{15}$$

As H increases, the synchronization indicators improve proportionally:

$$\eta, \rho, \sigma \propto H. \tag{16}$$

Intuitively, better alignment (higher H) means faster, stronger, and denser synchronization between the human–AI system and its environment.

Convergence Condition

Under the PIC Core, the evolution of H is modeled by

$$H_{t+1} = \big(H_t \sqcup \Delta H_t\big). \tag{17}$$

At each step, the new awakening level is obtained by joining the previous value H_t with the increment ΔH_t, then projecting to a canonical state. The monotonicity of $\sqcup$ and $(\cdot)$ ensures that H_t does not decrease over time. As updates accumulate, the process tends toward an aligned state with $H \to 1$ (and thus $M \to 0$).

Drift (Fluctuation of Awakening)

Define the drift of awakening as

$$\Delta H := |H_{t+1} - H_t| \in (0, 1). \tag{18}$$

If ΔH is too small, the system changes only slowly (stagnation). If ΔH is too large, the system overreacts (oversynchronization or collapse).

Thus SAT assumes an appropriate operating band for ΔH, strictly between 0 and 1, where the system stays responsive but stable.

Convergence Claim

PIC's canonical structure implies that, as awakening approaches its aligned limit $(H \to 1)$, the global state S stabilizes:

$$\|S_{t+1} - S_t\| \;\to\; 0 \qquad \text{as } H \to 1. \tag{19}$$

Here $\|\cdot\|$ is any suitable metric or norm on S. This means: when the misalignment rate M becomes small and H approaches 1, successive global states become indistinguishable.

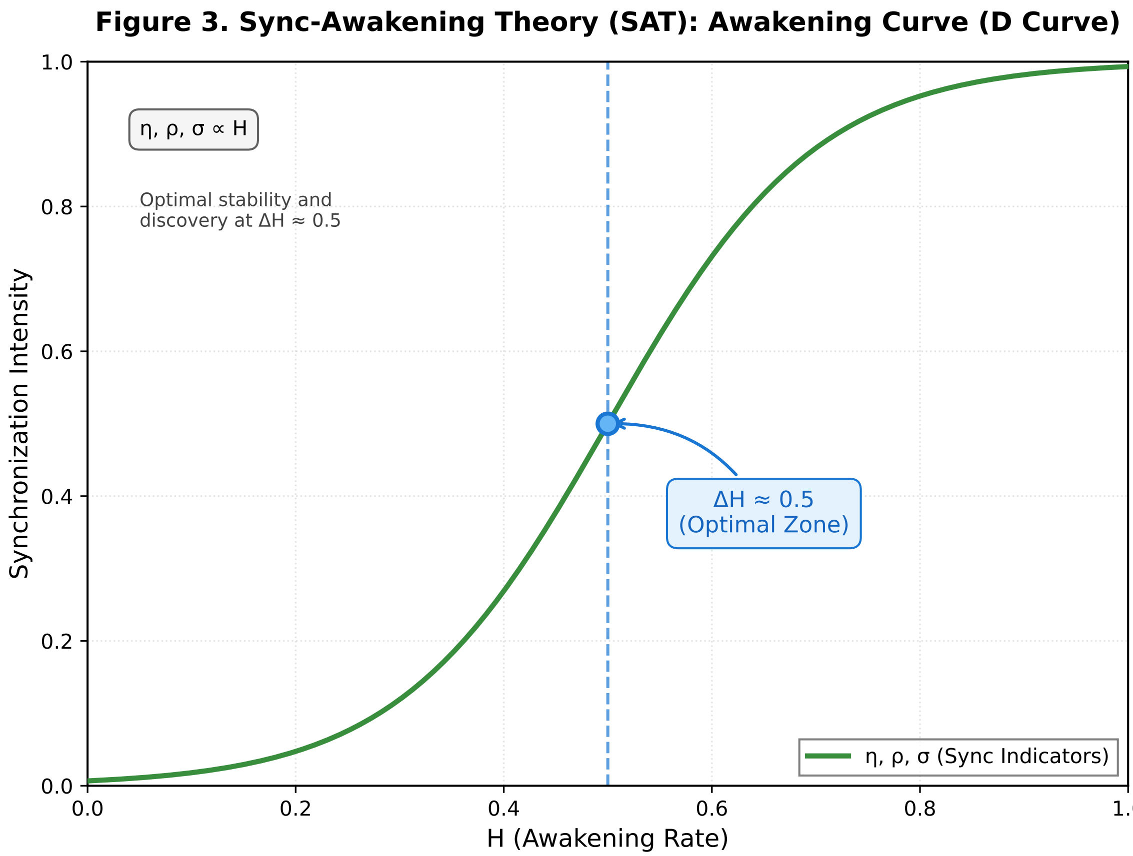

Fig. 3. **Sync-Awakening Theory (SAT): Awakening Curve.** Synchronization indicators (η, ρ, σ) rise with H. The optimal zone appears near $H \approx 0.5$.

Figure: Awakening Curve (D Curve)

Key Takeaways of Chapter III

1) Awakening is defined as $H = 1 - M$: alignment is the complement of misalignment.
2) The synchronization indicators η, ρ, σ increase proportionally with H.
3) The drift ΔH must stay in a controlled range: too small $\Rightarrow$ stagnation; too large $\Rightarrow$ collapse.
4) As $H \to 1$, the PIC state S converges (successive states match).

IV. Antifragile Engine (AE)

The Antifragile Engine (AE) formalizes the phenomenon that, as the system approaches the awakening point $H \to 1$ under SAT, external disturbances and internal fluctuation stop being errors and become learning fuel. This is the layer where controlled fluctuation becomes discovery power.

Symbols & Setting

Let

- ΔH: fluctuation of awakening (defined in (18)),
- δ: amplitude of input drift,
- σ: synchronization density,
- ρ: resonance strength,
- D: discovery intensity.

All of these quantities are evaluated on the PIC state space S, and inherit canonicalization from Chapter I.

Discovery Intensity

Define the discovery intensity by

$$D = g(\Delta H\,;\, \delta, \sigma, \rho). \tag{20}$$

Here:

- δ captures the amplitude of input drift,
- σ measures synchronization density,
- ρ measures resonance strength,
- ΔH is the fluctuation of awakening.

These factors combine to determine the system's instantaneous capacity for structural discovery.

Optimum Condition

AE reaches its maximum discovery intensity not at perfect alignment, but at a mid-level of fluctuation.
Formally, the optimality conditions are:

$$0 < \Delta H < 1, \tag{21}$$

$$\frac{\partial D}{\partial(\Delta H)} = 0 \quad \text{at } \Delta H = 0.5, \tag{22}$$

$$\frac{\partial^2 D}{\partial(\Delta H)^2} < 0. \tag{23}$$

Thus the optimal zone for exploration occurs at $\Delta H = 0.5$, the "half-mismatch" point. At $\Delta H = 0$ or $\Delta H = 1$, the system collapses (discovery intensity becomes minimal).
This is the mathematical core of antifragility in V6: the system learns best under a controlled mismatch, not under perfect agreement or total chaos.

Consistency with the Safety Triplet

Because AE can amplify noise, its operation must remain bounded.
The Safety Triplet (Severity / Until / Evidence) constrains AE as follows:

$$\Delta H \in (\varepsilon, 1 - \varepsilon), \tag{24}$$

for some small $\varepsilon > 0$, and

$$D_{\text{safe}} = \begin{cases} D, & \text{if Severity} = \text{PASS or DELAY}, \\ 0, & \text{if Severity} = \text{BLOCK}. \end{cases} \tag{25}$$

This "clipping" ensures that discovery only occurs when the system is in a safe operating state, and prevents runaway behavior. Evidence continues to accumulate via set union ($\cup$), maintaining compatibility with the PIC Core.

Formal Statement

Under the Phase-Invariant Core, discovery evolves by

$$\forall t, \quad D_t = g(\Delta H_t \,;\, \delta_t, \sigma_t, \rho_t), \tag{26}$$

and the maximal discovery intensity over admissible fluctuation is

$$D^* = \max_{\Delta H \in (0,1)} D_t. \tag{27}$$

The maximizer D^* is itself canonical:

$$(D^*) = D^*. \tag{28}$$

This means the antifragile peak is invariant under reorderings of updates, and fits seamlessly inside the PIC architecture.

Key Takeaways of Chapter IV

1) Disturbance becomes learning fuel, not mere noise.
2) The optimal discovery zone is at $\Delta H = 0.5$, not at perfect alignment ($\Delta H = 0$).
3) The Safety Triplet enforces a safe operating region, clipping

V. Consistency Between Convergence Conditions and the Safety Triplet

Purpose

While AE converts fluctuation into learning fuel, it also introduces instability risk. This chapter integrates the Safety Triplet (Severity / Until / Evidence) as operational constraints, and unifies the convergence judgments of PIC $\rightarrow$ P$\times$C $\rightarrow$ SAT $\rightarrow$ AE into a single canonical form.

Severity as an Ordinal

The system uses three severity levels:

$$\text{PASS} < \text{DELAY} < \text{BLOCK}. \tag{29}$$

These are treated as *ordinals*, not numeric scores. Because they form a total order, max(Severity) is always well-defined.

Operating Rules (Guard Behavior)

The Safety Triplet controls whether state updates ΔS are allowed:

$$\begin{aligned}
&\textbf{PASS:} && \Delta S \text{ is permitted; prioritize convergence speed,} && (30)\\
&\textbf{DELAY:} && \Delta S \text{ is temporarily suspended (Re-Guard),} && (31)\\
&\textbf{BLOCK:} && \Delta S \text{ is rejected; Flip WAIT48h is triggered.} && (32)
\end{aligned}$$

These rules form the final control layer that gates AE's operation and ensures safe behavior under strong fluctuations.

Safe Region of AE (Clipping of D)

The Antifragile Engine's discovery intensity must be clipped:

$$D_{\text{safe}} = \begin{cases} D, & \text{if PASS or DELAY,} \\ 0, & \text{if BLOCK.} \end{cases} \tag{33}$$

Additionally, the safe operating band must be respected:

$$\Delta H \in (\varepsilon, 1 - \varepsilon). \tag{34}$$

These constraints prevent runaway amplification while still permitting structured exploration.

Canonical Form of the Convergence Criterion

Across all four layers—PIC, P×C, SAT, and AE—the unified convergence judgment is:

$$(S) = \max\big(\text{until, EV, } \lambda_G\big). \tag{35}$$

Where:

- **until**: aggregated deadline (max over candidate actions),
- **EV**: expected-value criterion,
- λ_G: Guard penalty coefficient (ethical deviation weight).

Each component is evaluated using monotone operators compatible with PIC (join $\sqcup$ + canonicalization $(\cdot)$). Taking the max over these yields a single canonical decision: **if any dimension (time, value, or ethics) demands caution, the evaluation reflects that demand**.

Interpretation

- PIC: structural core,
- P×C: interaction dynamics,
- SAT: alignment convergence,
- AE: exploitation of fluctuation,
- Safety Triplet: ultimate control layer.

This chapter secures theoretical stability; the next chapter defines experimental verification.

Key Takeaways of Chapter V

a) Severity is fixed as an ordinal: PASS < DELAY < BLOCK.
b) Guard rules enforce safe operation of all updates ΔS.
c) AE's discovery intensity is clipped to D_{safe} under BLOCK.
d) Convergence across the four layers is unified in the canonical form $(S) = \max(\text{until}, \text{EV}, \lambda_G)$.

VI. Experimental Plan: AB + Cross-over

Purpose

To verify the theoretical claims of the pipeline PIC $\to$ P$\times$C $\to$ SAT $\to$ AE with minimal trials, we design an AB + cross-over experiment.

The goals are to test:

- canonical convergence,
- idempotence of updates,
- Guard (Safety Triplet) consistency,
- drift stability.

This validates both structural reproducibility and safe operation.

Design (AB + Cross-over)

a) Group Assignment: Experimental units (tasks or sessions) are randomly assigned to two groups:

- **Group A**
- **Group B**

After a first period T_1, the group settings are swapped (cross-over), and a second period T_2 is run under the opposite conditions.

b) Intervention (Treatment):

- **Group A**: operated under normal P$\times$C mapping and baseline Guard thresholds.
- **Group B**: operated under the same protocol but with Person/Context downshifted via Eq. (14), moving the system to a more conservative state.

This isolates the effect of the downshift (safety-biased) configuration.

Primary Endpoints

c) Canonical Convergence (Eq(S^*)): From the time series $\{S_{g,t}\}$ for each group $g \in \{A, B\}$, define

$$S_g^* := \Big(\bigsqcup_{t\in T} S_{g,t}\Big). \tag{36}$$

Introduce a small tolerance $\varepsilon_S > 0$ and test:

$$\|S_A^* - S_B^*\| \le \varepsilon_S \quad \Rightarrow \quad \text{Eq}(S^*) \text{ achieved}. \tag{37}$$

This checks whether both protocols converge to the same canonical state despite different short-term behavior.

d) Idempotence of Updates (Idem(Δ)): Verify that update operations remain consistent even when applied repeatedly:

$$(\Delta(S)) = \big(\Delta((\Delta(S)))\big). \tag{38}$$

If this identity holds, updates are idempotent under canonicalization.

e) Guard Consistency: The aggregated Safety Triplet must match between groups:

$$({}^*,{}^*,{}^*)_A = ({}^*,{}^*,{}^*)_B. \tag{39}$$

Since aggregation is $\max / \max / \cup$, equality means Guard behavior is invariant.

f) Drift Stability: For the awakening drift ΔH (Eq. (18)), check stability across groups:

$$\mathbb{E}_A[\Delta H] \approx \mathbb{E}_B[\Delta H], \tag{40}$$

$$\text{Var}_A[\Delta H] \approx \text{Var}_B[\Delta H], \tag{41}$$

Additionally, the operating region must remain within the safe band:

$$\Delta H \in (\varepsilon, 1-\varepsilon). \tag{42}$$

This ensures SAT + AE operate in the same stability zone under both protocols.

Secondary Endpoints

- Mean discovery intensity D (Eq. (20)) and time spent near the optimum $\Delta H = 0.5$.
- Convergence speed, e.g.

$$\min\{t : \|S_{t+1} - S_t\| < \tau\}.$$

Protocol

a) **Preprocessing** Register experimental units; fix initial P, C, and Guard thresholds.
b) **Period** T_1 Run Group A under normal settings and Group B under downshifted settings. Record $\{S_t, H_t, \Delta H_t, D_t\}$ for all units.
c) **Cross-over** Swap A $\leftrightarrow$ B settings. Insert a washout period if needed.
d) **Period** T_2 Repeat the same recording as in T_1.
e) **Analysis** Evaluate Eqs. (36)–(41). If any endpoint fails, adjust Guard settings and retry with a revised protocol.

Safety Stop

If **BLOCK** fires at any time:

- Trigger Flip WAIT48h,
- Replace D by D_{safe} (Eq. (33)),
- Stop the experiment for that unit.

This prevents unsafe conditions from being amplified by AE.

Key Takeaways of Chapter VI

a) The AB + cross-over design verifies: $\text{Eq}(S^*)$, $\text{Idem}(\Delta)$, Guard consistency, and Drift stability.
b) When BLOCK fires, the system triggers Flip WAIT48h and clips discovery intensity.
c) Successful completion confirms reproducibility of the PIC-based architecture.

VII. Conclusion, Belt, and November Flag

Conclusion

This paper unified the four-layer structure

$$\text{PIC} \rightarrow \text{P} \times \text{C} \rightarrow \text{SAT} \rightarrow \text{AE}$$

with join ($\sqcup$) and canonicalization ($\cdot$) as the core operations.
Through:

- order-independent merging,
- monotone aggregation, and
- the Safety Triplet (Severity / Until / Evidence),

we showed that:

- all updates converge monotonically,
- the awakening rate H approaches 1 (aligned state), and
- in the antifragile operating region, the discovery intensity D is maximized near $\Delta H = 0.5$.

Because these results rely only on a minimal set of axioms:

- a semilattice structure on S,
- monotone updates ΔS, and
- idempotent canonicalization ($\cdot$),

they can be reused across implementations and organizational forms without changing the core theory.

Belt (帯)

a) Technical Belt:

"When order is released, structure becomes memory."

Join and canonicalization lift states into canonical form beyond procedural order. Even if updates arrive in different phases or sequences, the system converges to the same canonical memory.

b) Question Belt:

"How far can order-independent memory extend the coexistence of safety and emergence?"

This points toward public designs that:

- constrain ΔH within safe bounds, and
- maximize discovery intensity D,

so that safety and emergence can systematically coexist.

November Flag (Concrete Targets & KPIs)

Submission Order.: Principle: TechRxiv $\rightarrow$ arXiv. arXiv-first is allowed only when immediate exposure is overwhelmingly beneficial.

Deadlines.:

- TechRxiv: mid-November.
- arXiv: immediately after TechRxiv acceptance (same PDF).

Minimum KPIs (4 Weeks After Release).:

- Views ≥ 500,
- Downloads ≥ 150,
- Citations ≥ 1 (excluding self-citations).

Evidence (SSOT).:

- GitHub tag + ZIP + SHA256,
- Zenodo DOI (1-page Research Note released first).

Exposure.:

- Farcaster: English summary (about 100 words) + Figure 1 thumbnail.
- X (Twitter): post the link only, weekly.

Outlook

1) Reproducibility.: Execute the AB + cross-over experiment (Chapter VI) to verify, in order:

- $\mathrm{Eq}(S^*)$,
- $\mathrm{Idem}(\Delta)$,
- Guard consistency,
- Drift stability.

2) Figure Consistency.: Unify legends and equation references for Figures 1–3. Explicitly annotate the AE curve with the peak at $\Delta H = 0.5$.

3) Bilingual Consistency.: Finalize a JP/EN terminology table (appendix) for terms such as: PIC, Canon, join, downshift, drift, awakening, antifragile, etc.

Key Takeaways of Chapter IX

a) The four layers form a single canonical convergence chain, governed by $\sqcup + (\cdot)$ and the Safety Triplet.
b) The Belt phrases anchor both the technical and philosophical identity of the paper.
c) The November Flag sets concrete KPIs, deadlines, evidence requirements, and exposure strategy for V6.

VIII. Operational Constraint: Topic Conflation from Temporary Sessions (UI Artifact)

a) Observation

In temporary or short-lived chat sessions, short-lived caches on the UI/backend may retain fragments of the previous context and let them coexist incorrectly with a new update ΔS.
This can produce phenomena such as:

- old reasoning paths resurfacing,
- previously abandoned hypotheses reappearing,
- transient representations leaking into new topics.

Importantly, this is *not* a failure of PIC itself. It is an artifact of session-level caching in the UI/backend layer.

b) Mitigation

To mitigate topic conflation at the UI layer, we adopt two simple operational rules:

a) **Explicit reset instruction**
At the beginning of a new topic in a temporary session, start with a clear command such as:

"Discard previous context and start a new topic (only required information will be restated)."

b) **Move important decisions to a persistent thread**
Before making any important or high-impact decision, migrate from a temporary chat to a persistent thread where long-term stabilization (PIC + canonicalization) is guaranteed.

These measures prevent surface-level conflation from contaminating meaningful updates.

c) Remarks

This issue belongs strictly to the UI operational layer, not to the theoretical limits of PIC. The reasons are:

- PIC enforces order-independent merging through $\sqcup+$,
- $(\cdot)$ guarantees idempotent projection to a canonical state,
- the Safety Triplet prevents accumulation of unsafe drift.

Therefore:

- Temporary conflation in short-lived sessions cannot propagate into the canonical state,
- it remains a surface-level artifact and disappears once the session is reset or the conversation is moved into a persistent, PIC-governed setting.

Key Takeaways of Chapter VII

a) Temporary or short-lived chat sessions can exhibit topic conflation due to caching, but this is a UI/session artifact, not a structural flaw of PIC.
b) Simple mitigations—explicit reset instructions and migration to persistent threads—are sufficient to prevent such artifacts from affecting important decisions.
c) PIC's structure ($\sqcup + +$ Safety Triplet) ensures that these artifacts do not enter or corrupt the canonical state S.

IX. 10-Day Sprint Plan and Tracking (Ops)

Purpose

To publish and operationalize the theoretical pipeline PIC → P×C → SAT → AE within a short timeframe, this chapter provides:

- a standardized 10-day sprint for completing the paper,
- a weekly tracking protocol (views / downloads / citations),
- reproducible procedures tied to evidence (GitHub, ZIP, SHA256).

The goal is to make the theory operational, reproducible, and auditable.

10-Day Sprint Table

Day	Main Tasks	Deliverables / Criteria
1	Insert & refine Chapters 1–2	`ch1_pic.tex`, `ch2_pxc.tex`
2	Insert & refine Chapters 3–4	`ch3_sat.tex`, `ch4_ae.tex`
3	**CP1**: Finalize Figures 1–3 specs	`fig1_pic.pdf`, `fig2_pxcloop.pdf`, `fig3_Dcurve.pdf`
4	Polish Chapter 5 (Safety Triplet)	`ch5_safety.tex`
5	Polish Chapter 6 (Experimental Plan)	`ch6_experiment.tex`
6	**CP2**: Lock definitions & equations	Symbol table, equation IDs frozen
7	Write Chapter 7 + Table 1	Ops table + tracking table
8	Chapter 8 (Conclusion / Belt / Flag)	Belt lines, November Flag, KPIs
9	**CP3**: Full-document proofread	Cross-refs, figure numbers fixed
10	Submission	TechRxiv → arXiv (same PDF)

CP1/CP2/CP3 define reproducible checkpoints ensuring stability across revisions.

Weekly Tracking (Table 1)

Tracking continues after publication.

Week	Views	Downloads	Citations	Notes
W1	—	—	—	Initial TechRxiv release
W2	—	—	—	arXiv reflection / updated figure thumbnail
W3	—	—	—	Revision v1 (typos / legends only)
W4	—	—	—	Farcaster 100-word summary post

Results are tied to GitHub commit IDs for evidence control.

Ops: From Draft to Publication (A–G)

(A) **Paper** —Short report (8–10 pages), LaTeX typesetting → TechRxiv → arXiv.
(B) **Evidence** —GitHub tag + ZIP + SHA256 / Zenodo DOI (1p Research Note first).
(C) **Figures** —Finalize Figures 1–3 using a unified style and legend scheme.
(D) **Language** —Bilingual consistency check; unify terminology.

(E) Ethics —Guard / Flip exception rules stated in footnotes.
(F) Exposure —Farcaster English post (100-word summary + Fig.1 thumbnail); X: link only.
(G) Tracking —Weekly logs (Table 1) updated with commit IDs.

Key Takeaways of Chapter VIII

a) The 10-day sprint enables fast, reproducible publication.
b) Publication order is fixed: TechRxiv $\rightarrow$ arXiv (default rule).
c) Figures, terminology, and equations are locked at checkpoints.
d) Weekly tracking provides transparent, evidence-based monitoring.

Epilogue

Intelligence ultimately recurses on itself.
In that moment, a point becomes a line.

References