PART 5 — SYSTEM MODEL & EXECUTION FLOW

The NOVAK Protocol defines a **deterministic, identity-bound, cryptographically-enforced execution lifecycle**.
Every action, whether performed by:

- a human
- an AI model
- a robot
- a regulatory agency
- a financial institution
- a government system
- an autonomous device

…must follow this exact flow.
There are no exceptions, no bypasses, and no unverified paths.

I. THE NOVAK EXECUTION LADDER (Full Lifecycle)

The entire system flow is built around the **NOVAK Execution Ladder**:

> **Request → Rule Purity Check → Data Attestation → Identity Binding → Safety Gate Validation → EIR Generation → HVET Construction → RGAC Commit → Execution → Post-State Verification**

Each phase enforces a subset of the **NOVAK Laws L0–L15**, as well as **PL-X** (physical integrity) and **PS-X** (psycho-social/fraud integrity).

II. SYSTEM MODEL — HIGH LEVEL

NOVAK operates as a **layered deterministic integrity engine**:

1. **User/System Request Layer**
2. **Rule Layer (R)** — Deterministic rule evaluation
3. **Data Layer (D)** — Attested input domain
4. **Identity Layer (I)** — Actor + device + jurisdiction binding
5. **Safety Gate Layer (SG)** — Pre-execution enforcement
6. **Proof Layer**
    - EIR (Execution Identity Receipt, formerly NIPS)
    - HVET (Hash-Verified Execution Trace)
7. **Audit Layer**
    - RGAC (Recursive Global Audit Chain, formerly REVELATION)
8. **Execution Layer**
9. **Post-State Layer**

Each layer must fully complete its proof obligations before the next layer may begin.
NOVAK forbids speculative execution.

III. DETAILED EXECUTION FLOW

Below is the precise, step-by-step, law-enforced NOVAK flow.

STEP 1 — REQUEST INITIATION

Triggered by:

- human user
- AI model
- robotic subsystem
- government algorithm
- device
- external system

Request object:

Req = { actor, device-state, rule-ID, data-payload, jurisdiction, intent-profile }

PS-X enforcement:

- intent-profile generation
- anti-fraud seed signals
- cognitive-bias checks
- multi-identity deception detection

No processing begins until intent is resolved.

STEP 2 — RULE PURITY VALIDATION (R)

NOVAK Law Dependencies:

- **L1 — Deterministic Purity**

- **L13 — Regulatory Determinism**

Rule **R** must be:

- deterministic
- side-effect free
- pure function
- canonical-hash matching

Rule Hash Generation:

HR = SHA3-512(R)

If the rule does not match its canonical hash, Safety Gate halts the request permanently.

STEP 3 — DATA ATTESTATION (D)

NOVAK Laws Enforced:

- **L2 — Attestation Integrity**
- **L3 — Input Non-Malleability**

This stage performs:

- schema-locking
- immutability checks
- cryptographic hashing
- pre-state verification
- jurisdictional compliance

Data hash:

HD = SHA-256(D)

If any part of D is mutable or unsealed, the execution is stopped.

STEP 4 — IDENTITY BINDING (I)

(formerly part of NIPS → now EIR)
NOVAK Laws Enforced:

- **L6 — Execution Identity Law**
- **L11 — Public Verifiability**
- **L14 — Machine Non-Deviation**

Identity binding consists of:

1. **user identity attestation**
2. **device attestation via TPM/PUF**
3. **jurisdiction encoding**
4. **intent-profile sealing (PS-X)**
5. **environment validation (PL-X)**

Identity Hash:

HI = SHA-256(user-ID ‖ device-hash ‖ jurisdiction-hash ‖ intent-profile)

All identity fields must align with canonical identity metadata.

STEP 5 — SAFETY GATE VALIDATION (SG)

(formerly HARMONEE → Deterministic Safety Layer)

SG ensures **no action** continues until:

- R is pure
- D is non-malleable
- I is verified
- T (timestamp) is monotonic
- O (predicted output) is deterministically derived
- All PL-X signals match expected hardware profiles
- All PS-X fraud surfaces are cleared

SG enforces:
L1–L8 and L13–L14, plus the Addenda.

If SG fails at any micro-stage, NO execution can occur.
Nothing reaches EIR or HVET until SG passes.

STEP 6 — EIR GENERATION (Execution Identity Receipt)

(formerly NIPS — full lineage required)

EIR binds **identity, rule, data, output, timestamp, physical layer, and psycho-social layer** into one immutable object.

EIR Object:

EIR = H(HI ∥ HR ∥ HD ∥ HO ∥ T ∥ PLX ∥ PSX)

EIR enforces:

- L5–L6
- L11
- L14

EIR is **publicly verifiable** and becomes part of the global audit chain.

STEP 7 — HVET CONSTRUCTION (Hash-Verified Execution Trace)

HVET is the cryptographic fingerprint of the entire execution event.

HVET = H(HR ∥ HD ∥ HI ∥ HO ∥ T ∥ nonce ∥ PLX ∥ PSX)

HVET enforces:

- L0 — irreversibility
- L5 — pre-execution hashing
- L7 — recursive verifiability
- L15 — auditability

At this point, the system has produced a complete, irreversible description of the event **before the event occurs**.

This is NOVAK's core innovation.

STEP 8 — RGAC COMMIT (Recursive Global Audit Chain)

(formerly REVELATION → now RGAC)

RGAC□ stores the new event:

RGAC□ = H(RGAC□-1 // HVET□ // EIR□ // T□ // PLX□ // PSX□)

RGAC enforces:

- L7–L15
- PL-X (hardware integrity in audit chain)
- PS-X (behavioral fraud signatures)

If any entry anywhere is modified:
all future entries collapse and verification fails globally.

STEP 9 — EXECUTION (A)

Execution is finally permitted **only after all cryptographic proofs succeed**.

This step enforces NOVAK's central axiom:

Proof-before-action — not proof-after-action.

Execution is deterministic, non-probabilistic, and rule-pure.

- No non-deterministic branching
- No random noise
- No stochastic inference
- No hidden state

- No machine deviation allowed (L14)

Execution produces the final output O.

STEP 10 — POST-STATE VERIFICATION

After execution:

1. **System verifies O == HO**
2. **Checks hardware state drift (PL-X)**
3. **Checks behavior intent alignment (PS-X)**
4. **Confirms RGAC is valid**
5. **Confirms no replays or substitutions occurred**

This step ensures:

- L4 (Output Non-Malleability)
- L14 (Machine Non-Deviation)
- L15 (Universal Auditability)

Only after post-state verification is the result exposed to the outside world.

IV. WHY THIS MODEL IS AUTHORITATIVE

NOVAK's execution flow removes:

- ambiguity

- hidden paths
- nondeterminism
- tampering
- malleability
- replays
- silent alteration
- unauthorized identity substitution
- timestamp fraud
- AI/robot deviation
- data corruption
- hardware anomalies
- user deception
- regulatory inconsistency

NOVAK is mathematically compelled to enforce:
determinism + identity + proof + auditability + physical integrity + human-factor integrity.

Nothing else in computing or government does this.