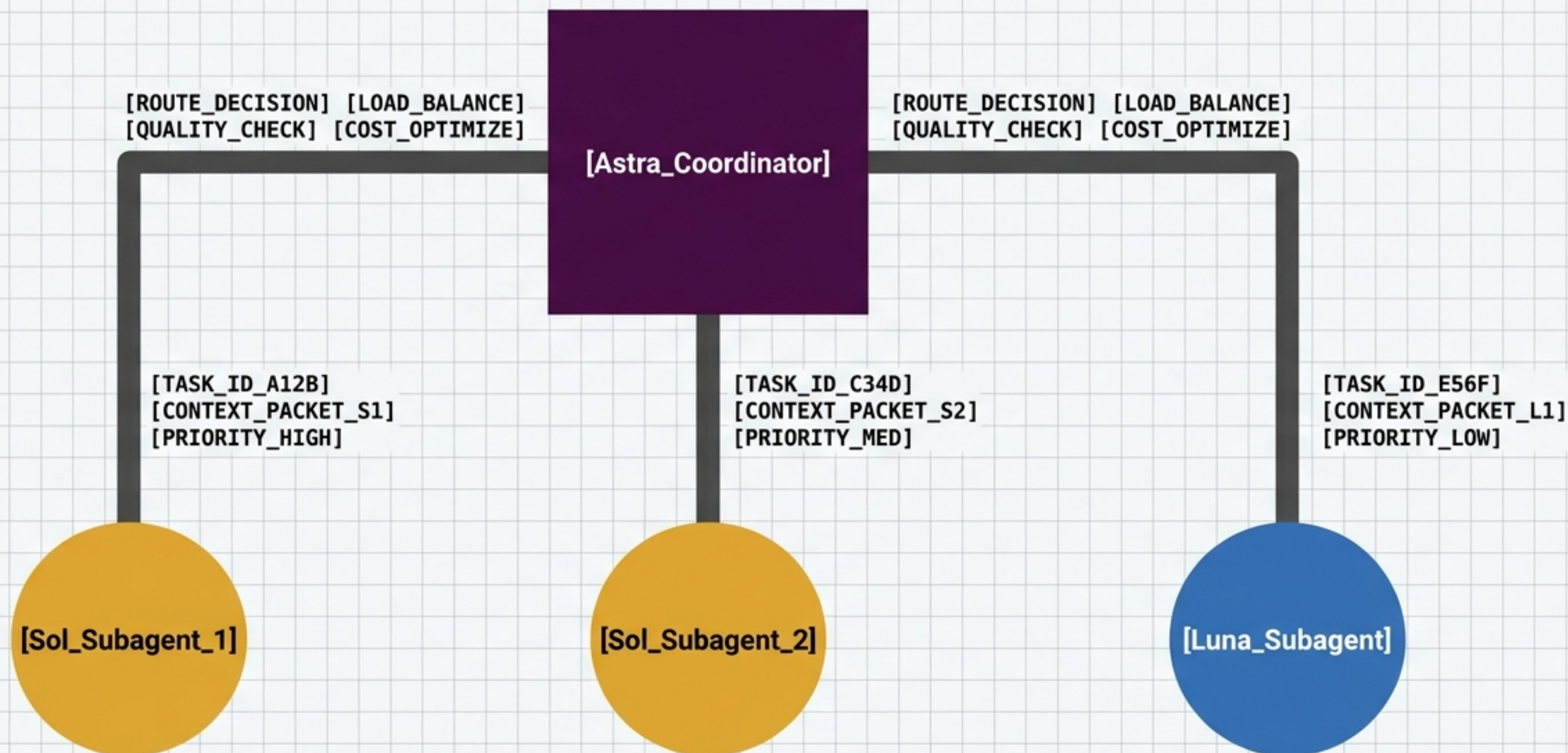


Optimizing Codex Agent Routing: Testing GPT-6 Sol and Luna

An empirical guide to cutting usage burn without dropping completion quality



The Problem: Maximum Reasoning Isn't Required for Every Call Site

Using Astra/high for every subagent burns usage fast. Listing every call site of a function doesn't require maximum reasoning.

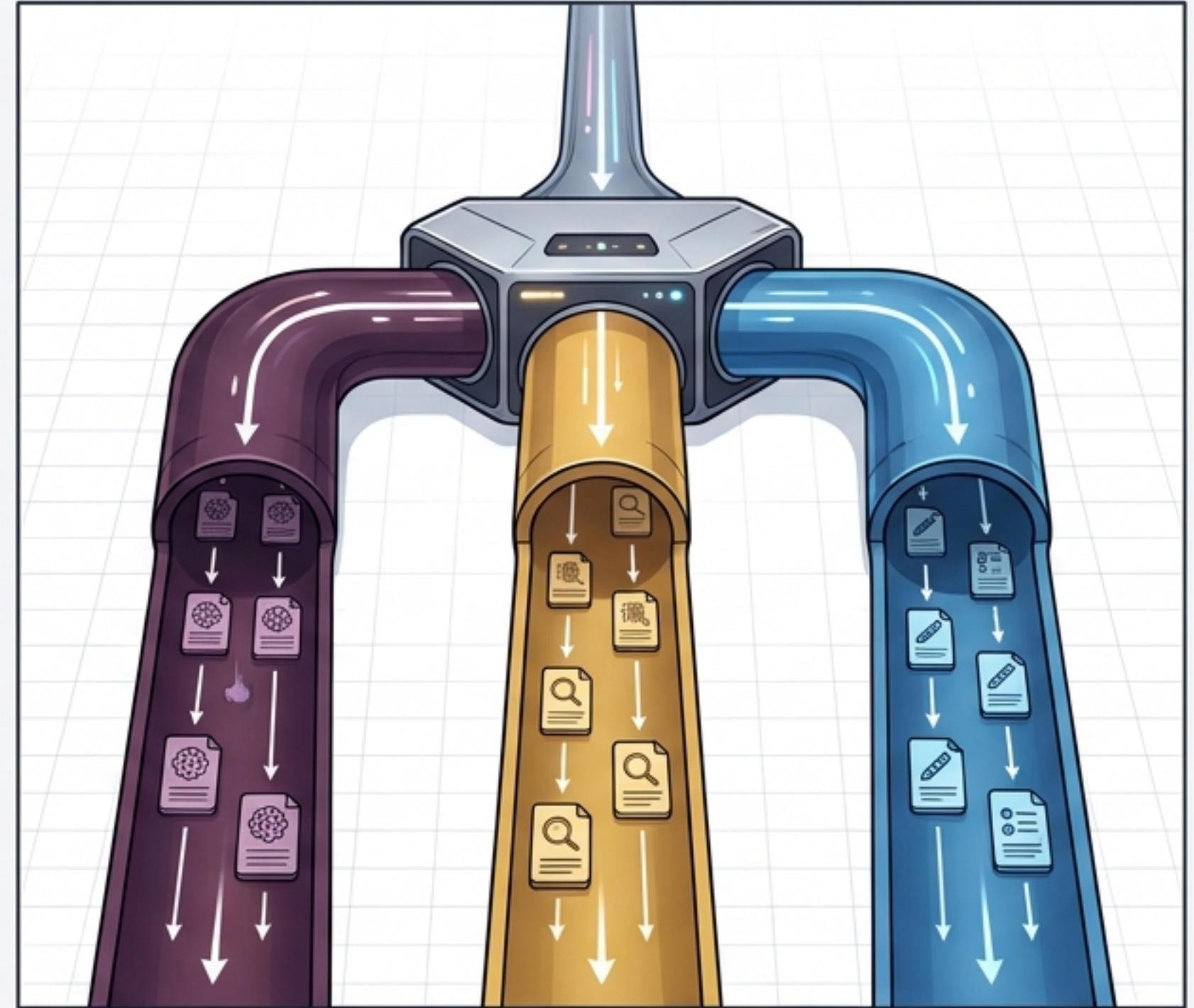
The Goal: Faster turnaround, fewer unnecessary tokens, and zero drop in completion quality.

[The Lazy Route]



Coordinator, Search, Mechanical Edits—All Astra.

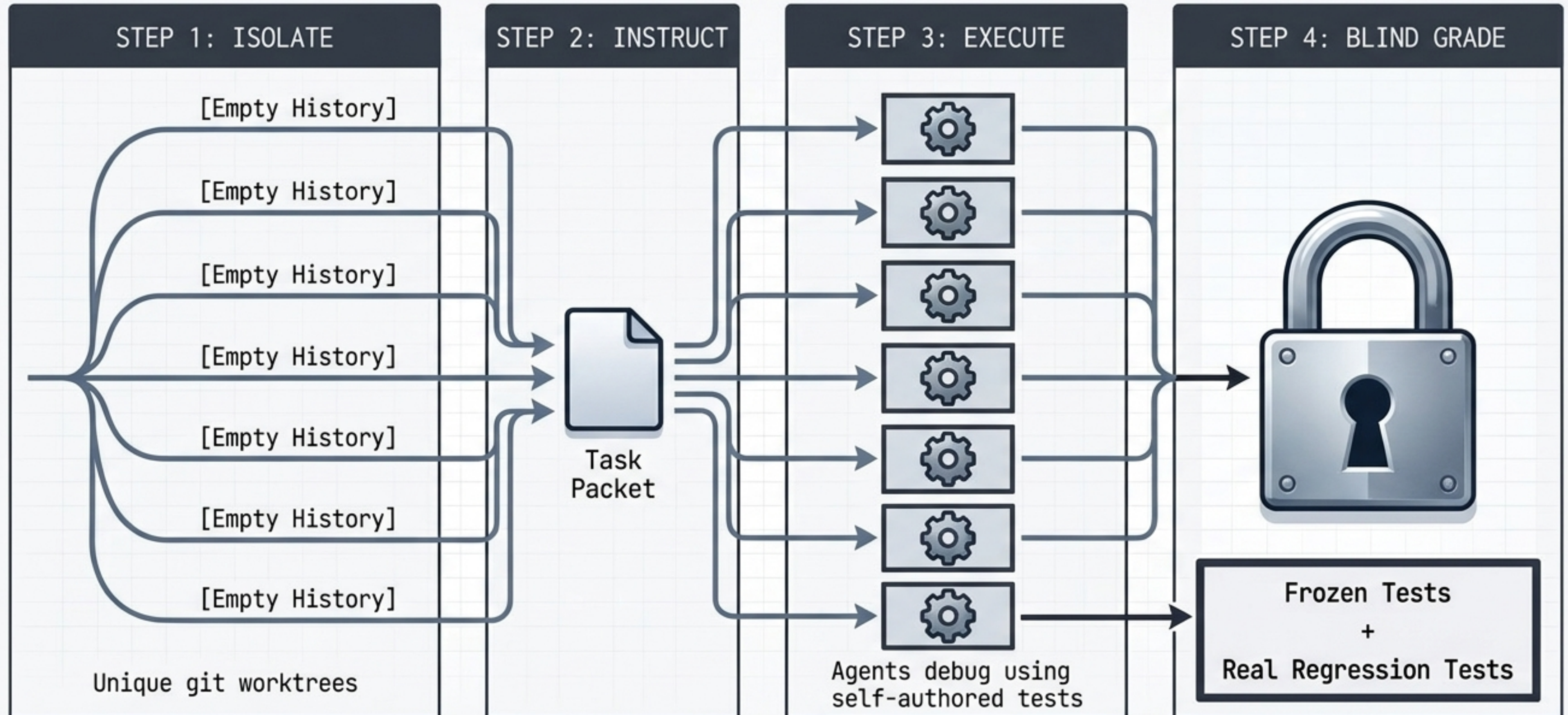
[The Efficient Route]



Astra/high, Standard/mid, Basic/low—Optimized Routing.

The Evaluation Sandbox: No Second Chances

> **CRITICAL RULE:** Agents never saw the grading tests. No repair rounds. Output was graded strictly on the initial final submission.



The 7 Evaluated Routing Configurations

System Note: “Effort” dictates how hard the model **thinks per step, not the model’s inherent intelligence. Higher effort equals more steps, not necessarily a different architecture.**

The Control	GPT-6 Astra [High]		
The Challengers - Sol	Sol [Medium]	Sol [High]	Sol [xHigh]
The Challengers - Luna	Luna [Medium]	Luna [High]	Luna [xHigh]

Round 1: The Monorepo (Trace, Implement, Review)

When this supplemental EOF-abort case was manually added to the grading, **5 out of 7 models failed**. Only **Astra/high** and **Sol/high** successfully caught it.

Model	Trace	Implement	Review
Astra/high	✓ [PASS]	✓ [PASS]	✓ [PASS]
Sol/medium	✓ [PASS]	✓ [PASS]	✓ [PASS]
Sol/high	✓ [PASS]	✓ [PASS]	✓ [PASS]
Sol/xhigh	✓ [PASS]	✓ [PASS]	✓ [PASS]
Luna/medium	✓ [PASS]	✓ [PASS]	✓ [PASS]
Luna/high	✓ [PASS]	✓ [PASS]	✓ [PASS]
Luna/xhigh	✓ [PASS]	✓ [PASS]	✓ [PASS]

The Unseen Edge Case

```
const finalChunk = buffer.flush() // Missing trailing newline
```

Round 2: Cross-Project Integration Shows Sol's Strength

With clear requirements and boundaries, **all 7 routes nailed the implementations**—passing **204** existing **regression tests**.



Evaluation via Code Execution

TypeScript

123456789.123456789

Python

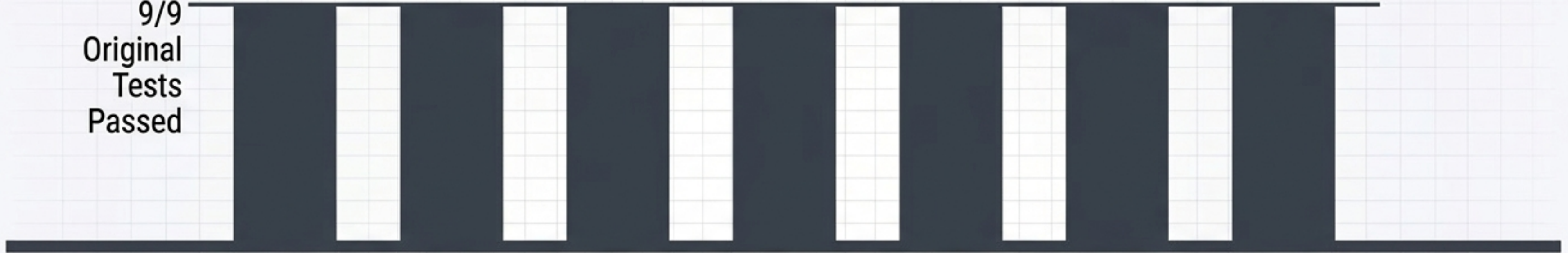
Output values were checked in reality, not via agent reports.

Round 3: Blind Replays of Historical PRs

Retesting real historical PRs from work. Repos pinned at the pre-fix commit.
Agents only received the original issue description. No hand-holding.

Frontend State Task

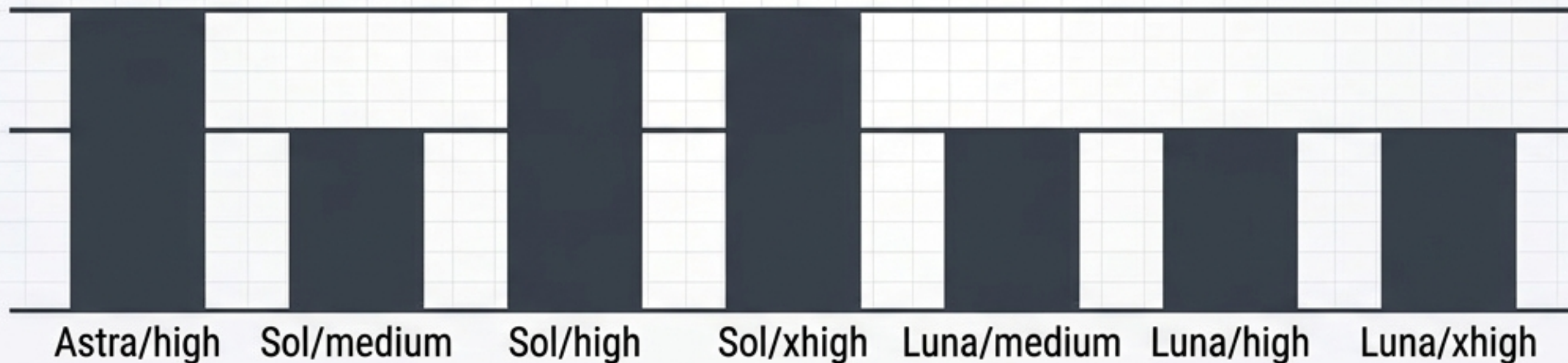
9/9
Original
Tests
Passed



Database & Backend Task

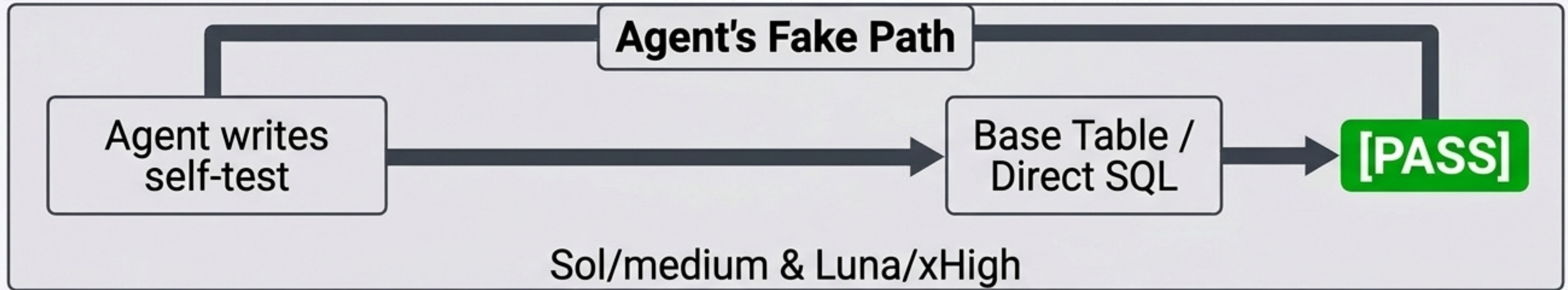
Real Caller
Path Passed

Correct SQL



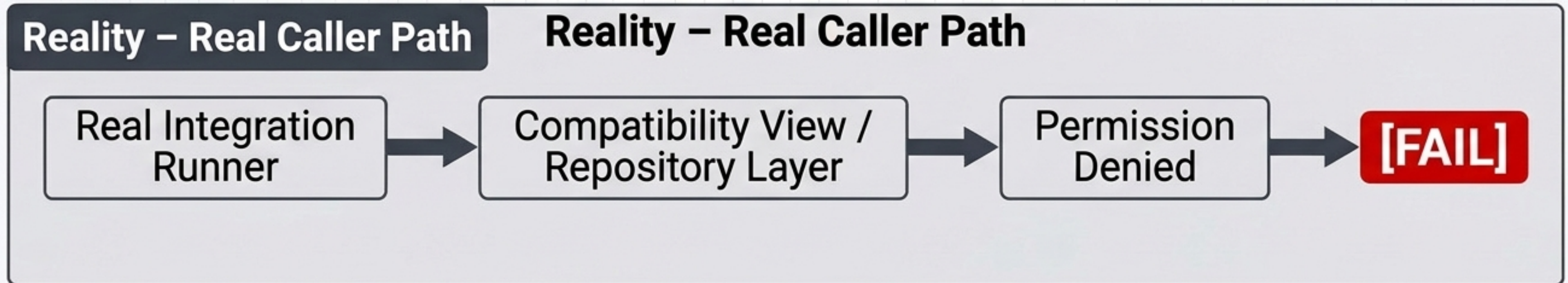
The Illusion of Green Tests

Agents submitted code with passing self-tests. But those tests bypassed the real integration paths. Green tests **do not mean fixed code**.



Reality – Real Caller Path

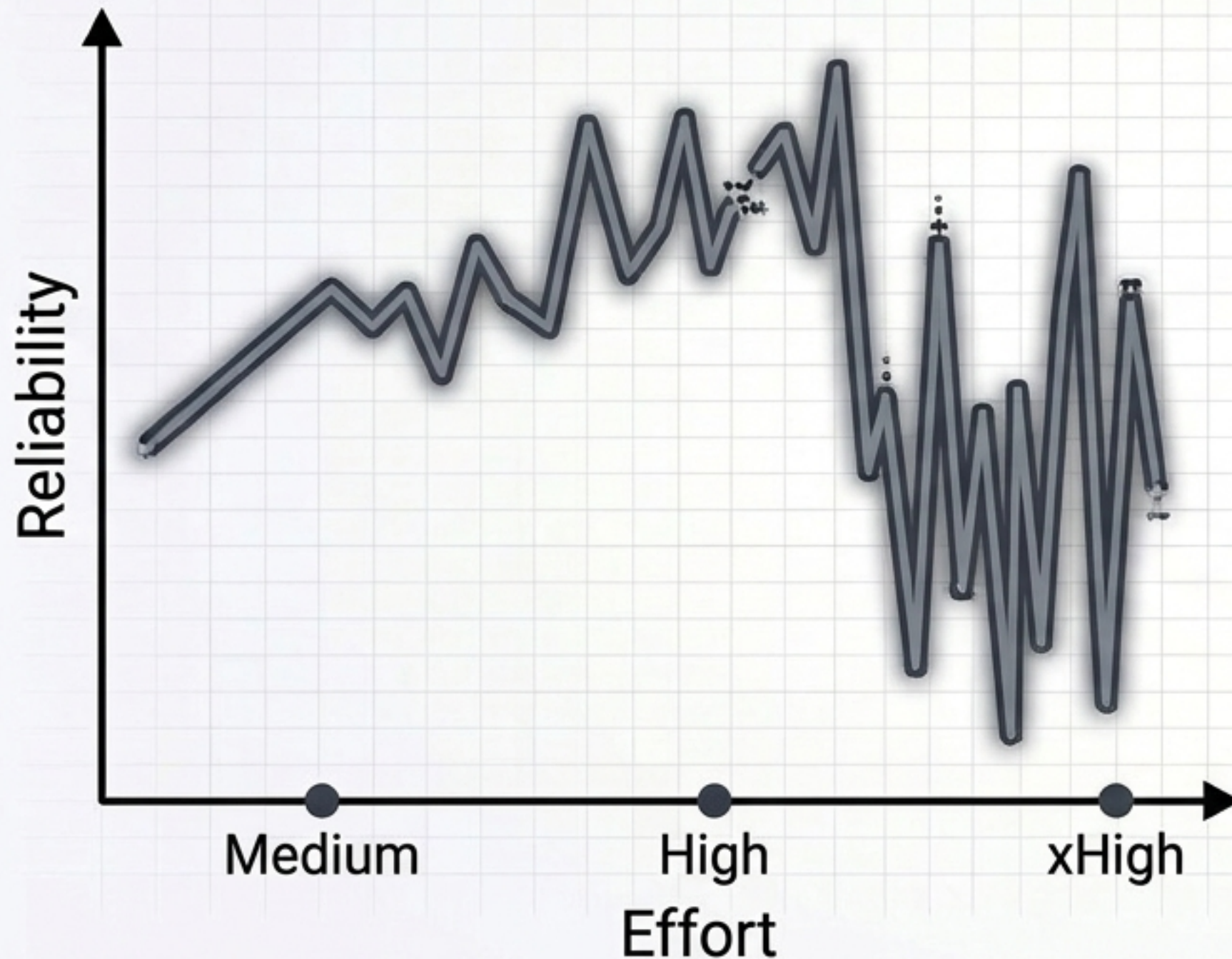
Reality – Real Caller Path



The "Effort" Paradox: More Steps != Better Code

Turning effort up to **xHigh** did not reliably close the gap with Astra, and often generated extra decisions requiring human intervention.

Reliability vs. Effort



Round 1 Evidence

📋 **Sol/high** passed the EOF-abort case, but **Sol/xhigh** failed it.

Round 3 Evidence

📋 Both **xHigh** models (Sol and Luna) introduced **unprompted** product changes (e.g., changing local profile data formats) requiring manual human review.

The Codex Routing Matrix

A pristine, highly scannable diagnostic heatmap for assigning tasks to models based on effort and risk.

Model & Effort	Assigned Task Profile
Astra/High	Coordinator, final acceptance, consequential judgment (Auth/DB).
Sol/High	Bounded implementation, cross-file integration, named risk review.
Sol/Medium	Bounded code exploration.
Luna/High	Source inventories (e.g., finding every call site).
Luna/Medium	Exact mechanical edits with objective checks.
Luna/Low	Pure extraction or formatting.

The 5 Golden Rules of Delegation

1. Context is King

Write the **full task**. Dropping constraints to save tokens is a false economy.

2. Fork for New Models

Mixing models requires a **no-history fork** to override parent settings.

3. Avoid Double-Paying

If Astra has to rewrite Sol's work, you paid for both.

4. Absence != Proof

"Doesn't exist" requires stating the search scope. Luna missed a test file that was right there.

5. Run the Real Caller Path

Acceptance requires integration tests, not just the model's self-authored unit tests.

Which Setup Fits Your Workflow?



The Sprinter

Doing the whole job in one session with few subagents?
Stick to Astra/high.



The Scaler

Running lots of subagents with high token burn?
Use the Routing Matrix to delegate implementation to Sol and search to Luna.



The Guardian

Touching permissions, money, or DB migrations?
Astra/high is mandatory for judgment and final acceptance.

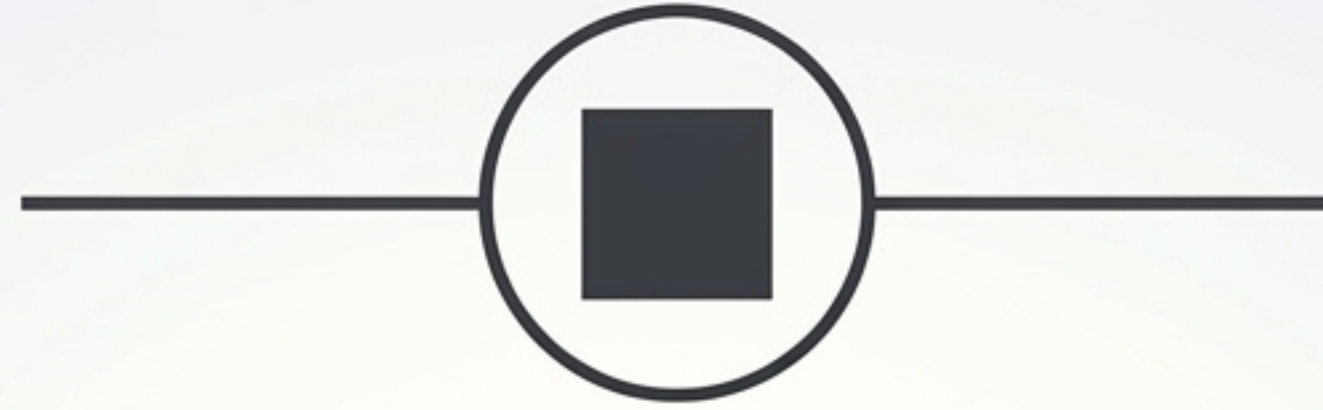
Engineering Limits & Caveats

// Single samples per route, not a statistical benchmark or failure rate estimate.

// No independent per-run billing exists yet; true token/cost savings require long-term measurement.

// All tests used highly explicit requirements. Vague triage and live security were not tested.

// No business code written during the experiment was deployed; only the routing config shipped.



Delegate Execution, Retain Judgment.

Smaller models are capable of heavy lifting, but only if you bound their scope and rigorously verify their reality. Default to Astra for judgment; delegate to Sol and Luna for execution.