PURETENSOR
INTELLIGENCE DISTILLED

Spend Is Width Times Turns

Where an orchestrating agent's tokens go, and what a diet moved

Paper: PT-R-2026-023 v1.0

Author: PureTensor Ltd

Date: 25 September 2026

Status: Published

Abstract

An orchestrating agent, a frontier language model in an agentic harness with tool access, runs our fleet day to day. Every assistant turn re-sends the entire conversation to the provider, which caches the prompt prefix between turns; the result is that spend scales roughly as context width multiplied by turn count. We instrumented 190 orchestrator sessions recorded between 5 August and 4 September 2026, finding that 94 per cent of input tokens were cache reads, two thirds of estimated cost came from re-reading the rolling conversation, and a quarter came from re-reading a fixed prefix of system instructions, memory index, tool schemas and skill catalogue. Sub-agent trees spawned on the same provider line carried a median of 8.2 million transcript tokens per session yet returned a median of 324 bytes to the orchestrator.

On 4 September we landed a set of interventions: a hook blocking whole-file reads above 40 kilobytes, removal of a duplicated per-prompt banner, a guard imposing width and turn fences with exploration budgets, and a rule routing sub-agents to worker seats on other providers. A second measurement on 10 September, covering 43 sessions from 4 to 10 September, showed the fixed prefix halved (90.4 thousand tokens to 46.4 thousand), file-read bytes fell 55-fold, sub-agent spawns fell from 54 to 1, and turns at the 90th percentile halved (255 to 134). Median context per turn fell only 11 per cent, and the 90th percentile rose; the conversation's share of spend increased from 65.7 per cent to 77.0 per cent because the prefix shrank around it, not because conversations grew.

Per-day spend moved from roughly 279 dollars to roughly 229 dollars, but the work mix differed between periods and the after window is one week, so this is not a controlled saving. The prefix was the easy lever and it is spent. The remaining cost is set by turn count and context width, the two quantities the diet did not directly control. A spend model of width times turns predicted the order of effect; measured order matched.

Threats to validity are plain: costs are list-rate estimates from transcript token counts rather than invoices, the after window is short, no session was run twice under both regimes, and the 90th-percentile context rise may reflect work-mix or may reflect the fence relocating effort into fewer, wider turns.

1. The Cost Geometry of a Re-sending Harness

The orchestrating agent is a frontier hosted language model with tool access: shell, file read and edit, browser, monitoring, task tracker, memory search. It directs cheaper worker seats for exploration and implementation. Every session's transcript is a streamed JSON-lines file recording, per assistant turn, the input tokens, cache-read tokens, cache-write tokens and output tokens.

The harness re-sends the whole conversation every turn. The provider caches the prompt prefix between turns, so each turn re-reads the accumulated context at the cache-read rate rather than the full input rate. In the baseline period, 94 per cent of input tokens were cache reads. This architecture means that a fixed prefix (system instructions, memory index, tool schemas, skill catalogue) is not a per-session cost but a per-turn cost: it is re-read on every turn the session takes.

Spend, then, is approximately width times turns. A session that runs wide and long pays quadratically in the conversation dimension and linearly in the prefix dimension. Controlling spend requires controlling both factors, with priority given to whichever dominates.

We wrote a measurement script over the transcripts and ran it twice: on 4 September 2026 over 190 orchestrator sessions recorded since 5 August (the baseline), and on 10 September 2026 over the 43 sessions recorded since 4 September (after the interventions). Costs are estimated at the provider's list rates: input at 10 dollars per million tokens, output at 50 dollars per million tokens for the frontier tier, cache reads at 0.1 times the input rate, and one-hour cache writes at 2.0 times the input rate.

2. Baseline Anatomy

The baseline covered 190 sessions over 31 days. The table below reports the orchestrator's shape.

Metric	Value
Sessions	190
Context per turn, p50 / p90 / max	250.1K / 564.3K / 996.1K tokens
Average first-turn cache write (fixed prefix proxy)	90.4K tokens
Fixed-prefix share of context mass	30.4%
Conversation share of context mass	69.6%
Turns per session, p50 / p90 / max	57 / 255 / 858
Tool-call turns / final-text turns	5,504 (78.8%) / 1,482 (21.2%)
File-read tool: calls, total bytes, average	928 calls, 65.8 MB, 70.9 KB
Sub-agent spawns (of which mid tier)	54 (36)
Sub-agent result bytes returned, p50 / p90 / max	324 B / 1.1 KB / 26.7 KB
Sub-agent transcript tokens per session, p50 / p90	8.2M / 116.6M

Estimated spend for the frontier tier over the baseline period was 8,664 dollars. The split by component was: conversation cache reads 44.9 per cent, conversation cache writes 20.8 per cent, fixed-prefix cache reads 19.9 per cent, output 10.2 per cent, fixed cache writes 4.0 per cent, uncached input 0.2 per cent. Two thirds of spend came from re-reading the rolling conversation; a quarter came from re-reading the fixed prefix; a tenth came from the model's own output.

File-read and shell tool results accounted for 84 per cent of all tool-result bytes injected into context. The fixed prefix of a fresh session was about 37 thousand tokens by component: project instructions 8.6 thousand, tool schema stubs 6.5 thousand, memory index 6.2 thousand,

skill catalogue 3.5 thousand, tool-server instructions 2.2 thousand, session-start hook 1.7 thousand, environment block 1.1 thousand. The first-turn cache write of 90.4 thousand also includes the first user prompt and any context injected with it, so it is a proxy for the prefix rather than a direct measurement.

Sub-agent trees dwarfed the orchestrator's own tokens. The median session carried 8.2 million tokens of sub-agent transcript against a median of 250 thousand per orchestrator turn. Sub-agents returned a median of 324 bytes to the orchestrator for those tokens. When sub-agents ran on the same provider line as the orchestrator, their cost appeared in the same bill; the tokens were not a saving but a transfer.

For contrast, the mid tier ran 237 sessions with context at p50 132.6 thousand and p90 349.6 thousand, and turns at p50 63 and p90 233. The small tier ran 27 sessions with context at p50 116.9 thousand and p90 437.8 thousand.

3. The Interventions

On 4 September we landed five changes.

A pre-tool hook blocks whole-file reads above 40 kilobytes; the file-read tool must be sliced. A per-prompt banner hook, which duplicated 115 tokens into every prompt, was removed. A guard hook on the read, search, shell and sub-agent tools imposes a width fence at 400 thousand tokens of context and a turn fence at 150 turns; beyond these thresholds exploration calls are blocked and shell calls are shaped. The guard also enforces an exploration budget of 300 kilobytes or 40 calls since the last sub-agent spawn. Noisy shell commands (cluster listings, log tails, commit logs, test runners) are routed through a log file with only the exit code and a slice returned.

A rule now prevents sub-agents from being spawned on the same provider line at all, with one exception for a fresh-context advisor. Exploration and implementation go to worker seats on other providers with a return contract of at most 2 kilobytes.

Doctrine was written down: verify by script rather than by reading artefacts; fewer turns beats a cheaper model; add nothing to the always-loaded prefix.

The guard was written by an implementation seat against a 48-case red battery. Using it on a live session found two defects the review had missed. First, the pool rule ran after the width fence, so under the fence a disallowed dispatch was shaped rather than blocked; a fence must never take a shaping branch ahead of an absolute rule. Second, the turn fence counted the whole transcript, so a session that compacted its context came back still fenced at 176 turns; a control that cannot be cleared by the remedy it recommends is a trap, not a fence. Both defects were corrected before the after period began.

4. After the Diet

The after period covered 43 sessions over 7 days.

Metric	Baseline (190 sessions)	After (43 sessions)
Context per turn, p50 / p90 / max	250.1K / 564.3K / 996.1K	221.7K / 690.6K / 967.7K
Average first-turn cache write	90.4K	46.4K
Fixed-prefix share / conversation share	30.4% / 69.6%	13.9% / 86.1%
Turns per session, p50 / p90 / max	57 / 255 / 858	47 / 134 / 724
Tool-call turns share	78.8%	59.8%
File-read: calls, total bytes, average	928, 65.8 MB, 70.9 KB	26, 1.18 MB, 45.5 KB

Metric	Baseline (190 sessions)	After (43 sessions)
Sub-agent spawns	54	1
Estimated spend (period total)	8,664 dollars (31 days)	1,605 dollars (7 days)

The fixed prefix halved, from 90.4 thousand tokens to 46.4 thousand. Its share of spend halved as well, from 23.9 per cent (cache reads plus cache writes) to 12.2 per cent. File-read bytes fell 55-fold. Sub-agent spawns fell from 54 to 1. Turns at the 90th percentile halved, from 255 to 134.

Median context per turn fell only 11 per cent, from 250.1 thousand to 221.7 thousand. The 90th percentile rose, from 564.3 thousand to 690.6 thousand. The conversation's share of spend rose from 65.7 per cent to 77.0 per cent (reads plus writes) because the prefix shrank around it, not because conversations grew.

The spend breakdown after the interventions was: conversation cache reads 59.9 per cent, conversation cache writes 17.1 per cent, output 10.8 per cent, fixed-prefix cache reads 9.7 per cent, fixed cache writes 2.5 per cent. The rolling conversation now dominated even more than before.

Spend component	Baseline	After
Conversation cache reads	44.9%	59.9%
Conversation cache writes	20.8%	17.1%
Fixed-prefix cache reads	19.9%	9.7%
Output	10.2%	10.8%
Fixed cache writes	4.0%	2.5%
Uncached input	0.2%	(not reported)

Per-day spend was about 279 dollars in the baseline (8,664 dollars over 31 days) and about 229 dollars after (1,605 dollars over 7 days). The work mix differed between periods and the after window is one week, so this is not a controlled saving.

For contrast, the mid tier after ran 24 sessions with context at p50 235.7 thousand and p90 658.3 thousand; its sessions grew, reflecting a different usage pattern. The small tier ran 7 sessions with context at p50 134.6 thousand and p90 180.9 thousand.

5. Threshold Calibration

On 10 September the fences were reviewed and kept at their original values: 400 thousand tokens for the width fence, 150 turns for the turn fence, 300 kilobytes or 40 calls for the exploration budget. The width fence sits between the after-period median (221.7 thousand) and the 90th percentile (690.6 thousand). The turn fence sits just above the after-period 90th percentile (134). The exploration fence bound in 2 sessions over the 7-day window.

Raising the fences would feed the component that is now 59.9 per cent of spend. The thresholds were therefore held.

Live guard errors over 24 hours: zero. A separate probe reports the fences daily and reddens on stale or missing data.

6. What the Numbers Say

The fixed prefix was the easy lever and it is spent. A quarter of the bill became a tenth. The remaining cost is set by two things the diet did not directly control: how wide the context gets

and how many turns the session takes.

The tool-result caps moved the width by 11 per cent at the median. The turn fence moved the tail; the 90th percentile of turns halved. But the 90th percentile of context rose, which suggests the fence may relocate effort into fewer, wider turns rather than reduce total work.

Sub-agents on the same provider line were not a saving but a transfer. Their transcripts were larger than the orchestrator's, and their returns were bytes. Moving them to other pools changed the bill's shape more than any prefix edit.

A spend model of width times turns predicts the levers in order: fewer turns first, narrower turns second, then prefix. Measured order of effect matched. The interventions that touched turn count (the turn fence, the exploration budget) and the intervention that removed sub-agents from the same provider line had the largest visible effects. The prefix trim was large in absolute terms but smaller than the conversation-driven costs.

7. What These Numbers Will Not Carry

Costs are list-rate estimates computed from transcript token counts, not invoices. Cache hit accounting follows the transcript's own fields; if the provider's internal accounting differs, the estimates will diverge from actual charges.

The after window is 7 days and 43 sessions against 31 days and 190 sessions. The work mix was not held constant. No session was run twice under both regimes. The per-day spend comparison (279 dollars to 229 dollars) is therefore not a controlled saving; it is a descriptive statistic of two periods with different workloads.

The 90th-percentile context rise (564.3 thousand to 690.6 thousand) may be work-mix, may be the fence relocating effort into fewer but wider turns, or both. The data do not distinguish these explanations.

The first-turn cache write is a proxy for the fixed prefix; it includes the first prompt. The 90.4 thousand and 46.4 thousand figures therefore overstate the prefix proper by the size of the first prompt, which varies.

The guard was tested against a 48-case red battery, but two defects were found only by live use. The battery's coverage is incomplete.

The measurement script and its findings files are retained and re-runnable. The daily probe continues to report. Claims that survive: the prefix shrank, file-read bytes fell, sub-agent spawns fell, turns at the 90th percentile fell, and the conversation's share of spend rose. Claims that do not survive without further data: that total spend fell by a controlled amount, that the context-width rise is benign, and that the fences are optimally placed.

8. Next Steps

The fences remain at 400 thousand width and 150 turns. The next experiment is to instrument the fence-hit rate per task category and test whether raising the width fence for a subset of tasks (those that require large artefact ingestion) improves completion rate without proportionate spend increase. A second line of work is to measure whether the context-width rise after the turn fence reflects relocated effort or a change in task mix; this requires tagging sessions by task type, which the current transcripts do not support.

PureTensor operates a sovereign AI infrastructure fleet (on-premises GPU compute, storage, and serving) run day-to-day by autonomous agents under human direction. This paper is PT-R-2026-023. Measurements were taken on 4 September 2026 (baseline, 190 sessions from 5 August to 4 September) and 10 September 2026 (43 sessions from 4 to 10 September). Raw artefacts are retained.