{ARA}

# Beyond Answer Accuracy: A CERRA Case Study of Language-Model Assistance in Transportation Analytics

Author names, order, and affiliations to be confirmed

Research manuscript draft — 2 October 2026

## Abstract

Analytical assistants must query intended data, preserve interface state, and respect evidential limits. We present a CERRA-specific case study using Maryland crash-data queries and historical Baltimore high-injury-network products. A frozen suite of 24 primary tasks and four controls, repeated three times, yields 1,848 scheduled application attempts across nine campaigns and thirteen model identifiers. Independent source-derived oracles, structured-plan checks, browser observations, and identity-blind AI-assisted narrative review evaluate the complete configured assistant. Local comparisons expose a quality–latency trade-off and a lighter configuration without established equivalence. The latest four-model API campaign matches medium reasoning, an 8192-token combined output cap, application inputs, and data. GPT-5.4, GPT-5.5, GPT-6 Astra, and GPT-6.1 Sol record 63/72, 68/72, 63/72, and 65/72 strict successes, with median request latencies of 3.91, 3.35, 2.54, and 3.24 seconds. GPT-5.5 has the highest observed score; its difference from GPT-5.4 is +6.94 percentage points with paired 95% interval [−2.78, +19.44]. All six exploratory unadjusted pairwise intervals include or touch zero, precluding a stable ranking or equivalence claim. Median-latency uncertainty is prespecified for this campaign; earlier intervals remain explicitly post hoc. A shared scope-reconciliation defect persists despite correct model-selected scope, and other differences involve supported-task refusals, ambiguous output contracts, and one unsupported narrative addition. Historical audit distinctions remain visible: an earlier Astra identity-registration amendment and a separate interrupted Sol outcome do not apply to the latest completed run. These bounded observations motivate separate measures of workflow success, pipeline attribution, state preservation, latency, and audit integrity. An English crash-analysis core and resource-reviewed asset-management candidates remain untested; no general transportation-competence or transfer claim is made.

**Keywords:** language-model evaluation; analytical assistants; tool use; human–AI interaction; transportation analytics; reproducibility.

## 1 Introduction

Natural-language interfaces can make analytical dashboards easier to approach, but they also change what correctness requires. A user who asks for the number of crash reports matching current filters may expect records outside the visible map to count. A request to clear the year should preserve every other selection. A question about next month's fatal-crash probability cannot be answered from historical counts alone. In each case, fluent language is compatible with an incorrect system outcome.

These issues are especially visible in transportation analysis, where similar terms can refer to different geographical units, missing observations should not be interpreted as negative evidence, and descriptive counts do not establish exposure-normalized risk or causal effects. An assistant may also change the analytical state from which a user draws subsequent conclusions. Assessing the response text alone can therefore miss both useful behavior and consequential failures.

We investigate this problem through a CERRA case study, not a general benchmark of interactive transportation analytics. CERRA is an existing web application that connects language-model assistants to Maryland crash data and Baltimore high-injury-network (HIN) source products. Its assistants use language models to select constrained operations and, for some tasks, narrate evidence. Shared application code validates plans, queries data, renders answers, and applies supported filter changes. The relevant unit of evaluation is the entire configured assistant, including these surrounding components. Questions about Baltimore-specific HIN products primarily test access to that implementation and its documentation. Other jurisdictions may use different methods, schemas, products, or no comparable HIN dataset; these tasks supply no evidence of transfer without a separate evaluation.

Our research questions are:

1. **RQ1:** How do configured assistants perform when success requires correct source use, task intent, narrative qualifications, and actual interface effects together?
2. **RQ2:** Within contemporaneous comparisons, how do strict task success, response time, and repeated-task consistency inform configuration choice?
3. **RQ3:** Which distinctions are lost when strict failures are summarized by a single accuracy number?

We contribute a documented evaluation design for a bounded CERRA workflow; an empirical account of thirteen model identifiers across nine campaigns; and a failure audit that distinguishes contract violations from practical state changes and evidential errors. Two local pairs, a matched two-model GPT follow-up, a three-model GPT extension, and a later four-model comparison including GPT-5.5 provide distinct contemporaneous comparisons. Historical deployments and a single-configuration API extension provide application context, with their distinct execution conditions retained. We do not pool them into a controlled leaderboard.

The study is deliberately limited. Its tasks were authored with implementation knowledge, narrative judgments are AI-assisted rather than expert-validated, and the model configurations differ in more than model family. Initial domain feedback identifies a mismatch between the suite's close coupling to CERRA and claims about broader transportation analytics. We address it by narrowing the empirical claim and separating a prospective benchmark redesign from the unchanged historical results. These limitations are not reasons to hide failed attempts or replace the frozen primary score.

# 2 Related Work

**Multi-dimensional model evaluation.** HELM argues for evaluating language models across standardized scenarios and multiple metrics rather than accuracy alone [1]. Our study applies this broad principle to a narrower application: correctness must be considered alongside interaction effects and latency. We do not reproduce HELM's coverage or assess its other dimensions, such as fairness and calibration.

**Agents in executable environments.** AgentBench evaluates language models through interactions with executable environments [2]. WebArena emphasizes realistic web environments and functional task outcomes [3]. These approaches motivate measuring the state that an interaction actually produces. CERRA differs from an unrestricted web agent: its model selects application-defined operations, while shared code performs validation, data access, and UI execution. The resulting evaluation concerns bounded application assistance rather than autonomous navigation across arbitrary websites.

**Tool use and repeated reliability.** $\tau$-bench evaluates tool–agent–user interaction using domain constraints and resulting database states, including reliability across repeated trials [4]. We likewise retain state and repeat-task outcomes, but use a fixed suite, three repetitions, and the directly observed number of tasks passing all three. We do not equate that count with a separately estimated general reliability metric.

**Application-relevant comparisons.** Kapoor et al. identify evaluation practices that can obscure practical agent performance and advocate comparisons relevant to downstream use [5]. Our response is to identify configurations and execution environments explicitly, preserve all attempts, and separate historical deployments from distinct contemporaneous pairs. We measure time, but not monetary cost or energy. The contribution is a transparent domain case study, not a claim that these evaluation principles originate here.

# 3 System and Evaluation Boundary

## 3.1 Two analytical workflows

The Maryland module supports queries over crash-report records, their temporal and geographic filters, ambient weather, contributor fields, and original or estimated map locations. The study uses frozen read-only indexes derived from the application's source snapshots. Public Maryland State Police dashboards describe data that may be revised or differ across reporting systems [6]; consequently, our oracles refer to the retained application snapshot, not a later live total.

The Baltimore HIN module exposes source-derived corridor, intersection, tract, and severe-injury records, together with documentary context. Tasks include rankings at specified corridor-length thresholds, source-ranked intersection selection, missing-value handling, and qualifications about scoring units or documentary disagreements. The six hashed exports cover ranked corridors at 0.5-, 1-, and 2-mile minimum lengths; the database Top300Intersections product; severe-injury and non-motorist segments; tract attributes with 2020 Census TIGER/Line geometry; and a source-version audit. Retained methodology notes identify the documentary source as "UMD/BCDOT Draft Final Report, September 2025." This is a retained source label, not a separately verified full report title. These are historical tests of faithful access to Baltimore-specific products. They are not a jurisdiction-independent HIN competency test and do not imply that another agency has equivalent inputs. They do not validate the underlying HIN methodology, crash-location estimates, or any countermeasure's effectiveness.

### 3.2 A constrained model–application pipeline

Figure 1 defines the system boundary. A request contains a question, current UI context, and where needed a canonical conversation history. The assistant routes the request through existing prechecks, a model planner, validation and any normal internal repair, deterministic data operations, and answer rendering. Some HIN responses include generated narration. Supported actions are applied by the browser; the evaluator inspects the resulting state.

Correct numeric output can arise because the model routed a question to the correct deterministic computation. It is not evidence that the model performed the arithmetic. Conversely, a bad final plan or answer can arise from shared reconciliation or rendering logic. Unless raw intermediate evidence isolates the cause, we report an application failure without assigning it exclusively to the foundation model.

## 4 Tasks, Oracles, and Metrics

### 4.1 Frozen task suite

The suite contains 24 primary tasks and four controls. Each configuration is evaluated three times per task. Primary categories are Maryland data queries (nine tasks), HIN source queries (five), HIN reasoning and limitations (three), supported UI filter changes (five), and ambiguous or unsupported UI actions (two). Three primary prompts are in Chinese; they are retained verbatim as historical experimental inputs, with no language-specific comparative claim from this small subset. They are not representative of an English-language evaluation for US transportation professionals. The prospective v0.2 design uses English only and is a separate, untested task version. The compact inventory in Appendix A identifies each task.

Nine of the 28 tasks are browser cases: seven primary tasks and two controls. The other 19 use the same authenticated API payloads as the UI. Thus each configuration–campaign observation contains 72 primary attempts and 12 controls, including 27 browser attempts. Nine campaigns retain all 1,848 scheduled application attempts: 1,584 primary and 264 control, or 594 browser and 1,254 API attempts. They include 264 identity-blind narrative-review records. One attempt has an interrupted, unretained application outcome and is accounted for separately in Section 5.5; 1,848 does not mean 1,848 completed responses. Primary request timing is recorded for 1,583 of 1,584 attempts. These attempts are not 1,848 independent questions or necessarily 1,848 individual model generations. Shared prechecks may answer without inference, and internal validation repair or separate narration may invoke inference more than once.

The controls are reported separately. MD16 tests a shared deterministic refusal; MD18, HIN01, and HIN11 are previously shown reference examples. None enters the primary score, primary latency, or paired model-ranking inference. A control answered by deterministic shared logic supplies no evidence of a model-quality difference.

### 4.2 Independent oracles and state preservation

Before inference, task records freeze the prompt, initial state, history, expected operation and scope, source-derived numeric or record-level expectations, required narrative propositions, and allowable browser effects. Follow-up tasks use a canonical prior context instead of the result of another benchmark attempt. The evaluated model receives the normal application request fields, not task IDs, gold plans, or scoring notes.

Maryland tasks include independently recorded read-only SQL anchors. For HIN, numeric oracles were computed in a separate Python pass over the frozen application JSON exports, without importing the application query, answer-rendering, reconciliation, or model code. This establishes computational independence from the evaluated path, not independent data collection or validation of the upstream extraction process. The oracle is independent of the response being judged. A returned plan that agrees with its own evidence is insufficient if both use the wrong county, temporal range, viewport scope, or source unit. Browser tasks also require the intended action to occur and unrelated state to remain unchanged. For an ambiguous or unsupported action, a successful refusal or clarification must preserve the current state.

### 4.3 Strict success and diagnostic outcomes

For model configuration $m$, primary task $i$, and repetition $r$, let $y_{mir} \in \{0, 1\}$ denote strict success. A pass requires the expected HTTP completion, correct selected-model identity, permitted operation and scope, required constraints and evidence, required narrative propositions, and expected browser effects. Missing an essential requirement fails the attempt. Optional qualifications are not promoted to requirements after observing an answer.

The strict rubric distinguishes contractual correctness from observed state changes. An extra action field can fail even if it redundantly repeats the current value. We therefore separately count *UI-state violations*: observed value changes that violate the frozen task policy among the 21 primary browser attempts per configuration. The five supported filter tasks also have their own 15-attempt success denominator. A 422 response

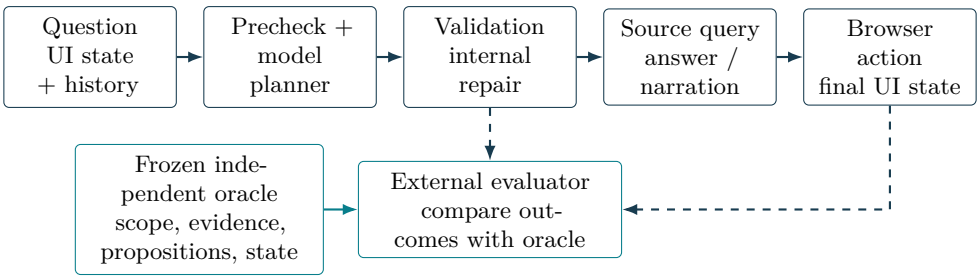



Figure 1: Evaluation boundary. The configured assistant includes the model, shared application logic, data operations, and browser effects. The oracle is available to the evaluator, not provided to the evaluated model. Some refusal controls terminate in shared prechecks.

is an application validation failure, not a transport outage. The paired MD17 violation applies a requested year change prematurely, before resolving the ambiguous geography; it does not imply that the year value was itself unrequested.

With $T = 24$ tasks and $R = 3$ repetitions, the primary score is

$$S_m = \frac{1}{T}\sum_{i=1}^{T} \bar{y}_{mi}, \qquad \bar{y}_{mi} = \frac{1}{R}\sum_{r=1}^{R} y_{mir}. \tag{1}$$

Every task has equal weight. Because all tasks have three repetitions, the score also equals primary passes divided by 72. We report the number of tasks passing all three repetitions as an additional directly observed consistency measure.

A secondary functional-equivalence analysis credits narrowly specified cases that satisfy a practical interpretation despite a strict mismatch. This analysis was post-hoc in the original campaign and inherited by subsequent campaigns. It never replaces strict success; reporting both exposes sensitivity to the interface contract.

## 4.4 Uncertainty and timing

For a contemporaneous pair $a, b$, the paired effect is

$$\widehat{\Delta}_{b-a} = \frac{1}{T}\sum_{i=1}^{T}(\bar{y}_{bi} - \bar{y}_{ai}). \tag{2}$$

We use 10,000 paired task-cluster bootstrap resamples, retaining all repetitions within each sampled task, and percentile 95% intervals. Marginal score intervals likewise resample each configuration's 24 task means. The single-configuration API extension has a marginal interval but no paired effect estimate. Historical campaigns use scheduling/bootstrap seed 20260918; the two- and three-model matched GPT campaigns use seed 20260930, and the ninth four-model campaign uses seed 20261002. The eighth and ninth campaigns report all three and all six pairwise contrasts, respectively, as exploratory intervals without multiplicity adjustment. Token sampling is not seeded. These unadjusted intervals describe suite-dependent uncertainty; repeated attempts are not treated as independent task draws, and the convenience sample does not justify population-wide significance claims.

Request latency measures the complete application request, including model loading, internal retries, and data access. Median and 90th-percentile (P90) values include every primary attempt with recorded request timing, including failures. Section 5.5 identifies the single interruption with missing completion time and its excluded latency observation; the primary success denominator is retained. Browser completion adds the final UI render and is reported separately. We also count strict successes completed within 30 seconds; for browser tasks, this criterion uses browser completion time. Controls remain excluded.

**Joint display of two marginal summaries.** The 1 October visualization added a descriptive, post-hoc uncertainty analysis for the earlier eighteen observations without rerunning inference or changing any success label; these intervals remain unchanged in the expanded display. The horizontal coordinate is strict primary task success, not answer-only accuracy; the vertical coordinate is median application-request latency in seconds. Previously reported marginal 95% task-cluster score intervals are retained unchanged. For each of those eighteen observations, a percentile 95% interval for the median resamples the 24 primary task clusters with replacement 10,000 times using seed 20261001. Each sampled task contributes all of its retained repetitions and their timings; failed attempts with measured timings remain included. The median is recomputed over those attempts in every resample, and the 2.5th and 97.5th percentiles use linear interpolation. Python `random.Random(20261001)` is restarted for each campaign; the same task-index draws are shared by all configurations within that campaign. The earlier matched Sol observation has 71 measured latencies; its missing interrupted timing remains missing, while its success denominator stays 72. Its timing interval is conditional

on recorded completion times and does not recover that unknown outcome. For the four new observations, the same median estimator and percentile procedure were instead specified prospectively before inference, using 10,000 shared task-cluster draws with seed 20261002 (Section 5.7). All 288 primary durations in that campaign are observed; the earlier missing Sol duration is not filled from the new run.

Horizontal and vertical whiskers are separate exploratory marginal intervals, without multiplicity adjustment. Their overlap is not a test of a paired difference. Their intersection is not a joint 95% confidence region, so we do not draw or interpret a confidence ellipse. A latency interval estimates uncertainty in the median under resampling of this task mix; it is neither P90 nor a range containing 95% of individual response times. No paired latency-difference interval is added. The visualization retains every campaign-specific observation rather than merging repeated model identifiers. Historical points differ in settings, routes, loading policies, and dates, and serve as descriptive context. Within any given set of conditions, the preferred direction is toward greater success and lower latency; the cross-campaign overview does not establish a global ranking or a deployment optimum.

### 4.5 Narrative review and audit history

Generated HIN narratives receive identity-blind, AI-assisted proposition review. The reviewer receives the frozen required and forbidden propositions and the answer, while model identities and aggregate results are withheld. Structural and semantic requirements must both pass. Across the nine campaigns, 264 narrative-review records are retained. This is single-reviewer AI assistance, not human expert validation or an inter-rater reliability study. The original campaigns' retained review receipts do not identify the reviewer model/provider, model snapshot, or tool version; this missing metadata limits exact adjudication replay. The API extension records a separate Codex reviewer context, but its exact backend model snapshot/version is not exposed. Context separation supplies procedural separation, not independent human expertise or independence of model training.

The September 29 local-pair review used the original notes, supplied evidence, and four anonymous historical judgments for calibration, without consulting identities, aggregate results, or external resources. The API-extension, lighter-model, and matched-API reviewers, including the ninth-campaign reviewer, received anonymized instructions, response packets, and calibration material, preserving the same scoring notes and historical interpretation. The reviewer did not inspect trial keys, model identities, current-run summaries, or network resources.

Audit history matters for reproducibility. The original campaign added an aggregation layer after freezing to enforce model-identity and browser-evidence requirements already present in the written protocol; original and protocol grades were preserved. Separately, the September 24 narrative packet initially omitted original scoring notes. Re-review of unchanged answers with complete notes and anonymous calibration changed 15 of 24 judgments. The revised packet was carried into the September 29 study. We report final audited scores while retaining initial artifacts. These facts support a frozen written rubric, but not a claim that every analysis implementation or review packet was preregistered without amendment.

## 5 Experimental Campaigns

### 5.1 Historical production observations

The September 18 campaign evaluates Qwen3 8B, Llama 3.1 8B, and Gemma 3 4B in production v36. Their engines and loading policies differ. Qwen3 uses a resident BF16 Transformers configuration with temperature 0.7, top-$p$ 0.8, and top-$k$ 20. Llama and Gemma use Ollama Q4_K_M packages with temperature 0.1 and cold-loading behavior. These are practical configurations, not a controlled model-size comparison.

The September 23 campaign evaluates GPT-4.1 mini, GPT-5.4 mini, and GPT-5.4 through the OpenAI Responses API in production v48. Planner calls request JSON-object output with an additional JSON-only instruction, per-call output-token budgets, a 60-second model-call timeout, and storage disabled. Temperature, reasoning effort, and inference seed are not explicitly supplied. Provider aliases are not immutable weight snapshots.

The September 24 campaign evaluates Qwen3 14B and Gemma 4 12B. Additive provenance identifies the actual v51/v52 deployments as sharing the same application archive and environment revision; it supersedes the initial v50 manifest label. An independent source audit preceded inference, although one machine-readable source receipt was captured shortly after the first attempt. The paper preserves these details instead of presenting a cleaner but inaccurate chronology.

All campaigns retain the same task objects and written acceptance constraints. Campaign-specific scheduling/bootstrap seeds and the later interruption-handling amendment are identified separately. Calls are sequential and reset to frozen context. Three-model campaigns rotate first, middle, and last position within tasks. Two-model campaigns alternate model order, balancing positions over the schedule but not perfectly within each three-repetition task. No external retry replaces a failed attempt. Prompts, generation settings, oracles, and written strict criteria remain unchanged during inference; the audit-layer changes are described above.

## 5.2 Contemporaneous local Qwen comparison

The September 29 study compares qwen3:14b with qwen3.8:27b-q4_K_M. It reuses the historical application base, frozen data, and all 28 task records, with local routing and timeout additions. Five Maryland database hashes and six HIN JSON hashes match the retained historical source snapshot. The nine Maryland numeric/source oracles were independently replayed. The two package digests are retained in the paper's provenance inventory.

Both configurations use Ollama 0.34.3, Q4_K_M quantization, temperature 0.1, an 8192-token context setting, disabled thinking, and `keep_alive=0`. They share an RTX 5090 with 32 GB memory; the resident production Qwen3 8B and camera processes remain active. Partial CPU offload for Qwen3.8 is observed in a connectivity smoke check. Speculative decoding is not explicitly configured; runtime option resolution implies it is disabled, but it is not separately measured per trial.

Each repetition uses a seeded task shuffle. Within a task, model order alternates according to the fixed case index and repetition index. Each model is first in 42 of the 84 task–repetition pairs. The run retains all 168 outcomes, 54 browser records, and 24 narrative reviews. It executes a local compiled Worker rather than traversing the public production-site network path.

Architecture, parameter count, memory residency, and offload differ between the two configurations. Consequently, the comparison estimates a practical configuration difference under shared conditions. It cannot isolate a release-generation effect, and its latency should not be interpreted as a controlled baseline for earlier production measurements.

## 5.3 September 29 API extension

A fifth campaign evaluates gpt-6.1-sol, displayed as GPT-6.1 Sol, through the OpenAI Responses API. It uses an isolated compiled copy of the same historical application base and the retained source snapshots. The preflight audit matches five Maryland database hashes, six HIN export hashes, and the nine original Maryland oracle payloads; structural-grader parity is checked against 588 historical attempts. Only the model catalog and OpenAI transport receive the new routing and configuration. Planner and narration prompts, validators, internal repair, task objects, source oracles, and written strict scoring remain unchanged.

The configuration explicitly requests medium reasoning effort, an 8,192-token total output budget, a 60-second model-call timeout, and disabled storage. The output budget includes both reasoning and visible output; it is not an 8,192-token answer-length target. Existing concise-answer prompts remain in force. Temperature is not supplied, and planner calls retain JSON-object output. The provider route is the regional US Responses endpoint. Provider receipts account for 93 calls during scored application attempts, all reporting the requested gpt-6.1-sol alias; separate smoke calls are excluded. This alias is not an immutable weight snapshot.

This is a single-configuration extension: 24 primary tasks and four controls are each evaluated three times, with the inherited seeded task shuffles. All 84 application outcomes, 27 browser records, and 12 narrative reviews are retained and verified. It has no contemporaneous comparator and produces no within-campaign paired model-difference estimate. Request latency includes the frozen application's work and remote API calls, while browser completion also includes the resulting interface update. These timings have a different execution path from both earlier public production requests and local Ollama inference. Explicit reasoning effort and the combined output budget also differ from the historical API configurations. Cross-campaign score and timing differences are therefore descriptive observations of practical configurations, not equal-compute comparisons or isolated model-generation effects.

## 5.4 Lighter local configuration and a fresh comparator

A sixth campaign pairs qwen3.5:9b-q4_K_M with a fresh gemma4:12b observation in an isolated application built from the same historical base. Both use Ollama 0.34.3 on the shared RTX 5090 (nominal 32 GB; reported capacity 32,607 MiB). The 11 source snapshots, nine Maryland oracles, and 588 historical structural grades pass the retained parity checks. Tasks, contexts, scoring rules, and the alternating two-model schedule remain unchanged. All 168 attempts, 54 browser records, and 24 identity-blind narrative assessments are complete; independent integrity and aggregate-arithmetic checks pass. The run occurs at 01:45–02:01 UTC on September 30, corresponding to September 29 in Maryland. It has its own manifest and does not replace the September 24 Gemma observation.

Both packages use Q4_K_M quantization, temperature 0.1, an 8192-token context, disabled thinking, and `keep_alive=0`. Package defaults differ: Gemma uses top-$k$ 64 and top-$p$ 0.95; Qwen uses top-$k$ 20, top-$p$ 0.95, and presence penalty 1.5. Token sampling is not seeded. Request times include cold model loading and failed attempts. Shared GPU workloads remain active; warm sessions, exclusive-GPU operation, and concurrency are not evaluated. Thus this remains a practical configuration comparison, not an isolated architecture or parameter-count experiment.

The application attempts generate 99 Gemma and 93 Qwen model calls; six MD16 controls terminate without inference. Supplementary Ollama residency sampling covers all 99 Gemma calls and 77 of 93 Qwen calls.

Reported `size_vram` is 7.80 GiB for Gemma and 5.34 GiB for Qwen in the retained samples, with no CPU offload observed there. Sixteen Qwen calls lack residency samples; their placement is unknown. These values describe sampled model residency, not continuous peak GPU memory, whole-device usage, or minimum hardware requirements. They must not be compared as though they were the same metric as historical model-weight buffer allocations. Per-call median load times are 2.37 seconds for Gemma and 2.57 seconds for Qwen; they are distinct from application-request medians and are not additive decomposition estimates.

## 5.5 Matched GPT API follow-up

A seventh campaign compares gpt-5.4 and gpt-6.1-sol in one isolated copy of the historical CERRA application. Both use the regional US Responses endpoint, medium reasoning effort, an 8192-token combined reasoning/visible-output cap, a 60-second model-call timeout, disabled storage, no supplied temperature, and JSON-object planner output. The requested GPT-5.4 alias resolves to the provider-reported gpt-5.4-2026-03-05 identifier; Sol receipts retain gpt-6.1-sol. Neither provides a locally retained weight snapshot. Provider GPU memory and internal compute are not exposed; matching request parameters is not evidence of equal realized compute. These matched settings apply to the isolated experiment and do not change the production GPT-5.4 configuration.

This follow-up keeps the historical 28-task suite, including its Baltimore HIN and Chinese inputs. It does not evaluate the proposed English v0.2 core. Tasks and three repetitions are unchanged; model order alternates under a fresh prespecified seed of 20260930. Initial provider-call messages are byte-equivalent across the 81 task–repetition pairs that invoke inference; the three MD16 control pairs terminate without a provider call. Subsequent repair or narration can depend on earlier outputs. The retained receipts attribute all 189 provider calls from scored attempts.[1] These checks tighten the comparison relative to historical observations collected with different settings and dates, while leaving those observations intact.

**Interrupted outcome and analysis amendment.** Collection retains 168 settled application attempts: 167 responses with HTTP 200 and one unknown application outcome. After 84 responses had been retained, a SIGTERM interrupted the runner during Sol's repetition-2 MD07 attempt. Its provider call returned HTTP 200, but the application response and completed end-to-end time were not retained. The signal's cause is unknown. The event is recorded as an infrastructure interruption, not evidence of a model-caused error; the administrative status 0 is not an HTTP response code. That attempt is never retried. The runner resumes only the 83 not-yet-started attempts, preserving the manifest, tasks, application, settings, and graders.

A retained post-interruption amendment keeps the unknown outcome as a primary non-success, preserving 72 scheduled primary attempts per model and the frozen equal-task primary estimator. Request-latency summaries exclude its missing completion time rather than substitute a timeout, provider duration, or settlement time; the measured primary denominators are 72 for GPT-5.4 and 71 for Sol. Two explicitly secondary checks accompany the primary estimate: a complete-pair sensitivity drops both repetition-2 MD07 observations, leaving 71 attempts per model, and an arithmetic best/worst bound assigns the single unknown Sol outcome either zero or one success. The complete-pair statistic weights retained attempts equally; its paired task-cluster bootstrap keeps the two-observation MD07 cluster and three-observation clusters otherwise. It is a changed estimand, not a replacement for the equal-task primary result. The one-attempt bound is not a confidence interval and does not recover the lost outcome. These post-interruption additions are disclosed rather than described as an unamended preregistration.

## 5.6 Three-model matched API extension

An eighth campaign adds gpt-6-astra, displayed as GPT-6 Astra, alongside fresh gpt-5.4 and gpt-6.1-sol observations. It uses an isolated copy of the same historical CERRA application and the unchanged 28-task suite. The model catalog and API routing add Astra; the tasks, prompts, data, independent oracles, scope reconciliation, application validators, and UI behavior remain frozen. A post-run model-registration amendment to the HIN grading implementation is disclosed below; the final grader is not byte-identical to the initial frozen implementation. This extension does not test the proposed English v0.2 core or overwrite any earlier GPT observation.

All three configurations request medium reasoning, an 8192-token combined reasoning/visible-output cap, a 60-second model-call timeout, disabled storage, no temperature parameter, and JSON-object planner output through the same regional US Responses endpoint. The frozen expected response identifiers are gpt-5.4-2026-03-05, gpt-6-astra, and gpt-6.1-sol, respectively. Provider receipts are checked against these identities; none is an independently retained weight snapshot. Provider GPU type, memory capacity, actual model VRAM usage, and internal compute are not exposed. Matching these request settings therefore does not establish equal realized compute.

[1] Non-scored connectivity smokes are excluded from the experiment. Raw-response retention is incomplete for some preliminary smoke calls; this does not remove any scored provider-call receipt or supply the interrupted application outcome.

The schedule contains 252 attempts: 216 primary and 36 controls, including 81 browser and 171 API attempts. Each case–repetition forms an adjacent three-model triplet. A prespecified Latin schedule places each model in each position once per task across the three repetitions; all six model-order permutations occur fourteen times across the 84 triplets. Within one repetition, each model occupies each position eight to ten times, rather than perfectly equal counts. Task order is independently shuffled for each repetition under seed 20260930. The nine MD16 control attempts terminate through the unchanged deterministic precheck and supply no inference evidence. Application-internal repair and narration remain active; there are no harness retries or mid-run tuning.

The prospective interruption policy retains any started attempt without an application outcome as an infrastructure non-success, never reconstructs the missing outcome from provider output, and never reruns that attempt. Any resume starts only not-yet-started attempts. The primary score keeps all scheduled attempts; latency omits unmeasured completion times with explicit observation counts. If infrastructure loss occurs, the prespecified secondary analysis removes the affected case–repetition triplet from all three models, uses attempt-weighted paired task-cluster resampling with the resulting unequal cluster sizes, and reports unknown-outcome bounds separately. These analyses do not replace the primary result or attribute infrastructure loss to model quality.

Thirty-six identity-redacted narrative records underwent independent AI-assisted review against the frozen scoring notes and calibration before aggregate release. This is not human domain-expert validation. All three pairwise score differences use 10,000 paired percentile task-cluster resamples over the 24 equally weighted task means, with seed 20260930. Every pairwise interval is exploratory and unadjusted for multiple comparisons; neither a universal winner nor equivalence is inferred from these intervals. The known MD01 scope-reconciliation defect and MD04 annual-row rubric ambiguity are retained and diagnosed separately.

**Post-run model-registration amendment.** The initial HIN grader contains a hard-coded model-identity allowlist that omits Astra, although Astra is explicitly registered in the frozen experimental manifest and its provider identity is checked. It consequently rejects Astra outcomes at the identity-registration check. After collection, a narrow amendment registers the already scheduled identifier in this allowlist. The original grader, original grades, and the pre-amendment aggregate are preserved as audit records. These registration rejections are not evidence of Astra's task performance. The final reported scores use the amended implementation, not the initial registration-defective aggregate. No inference is repeated and no answer, task, oracle, permissible action, evidence requirement, or narrative judgment is altered by this amendment. The amended model registration implements the existing experimental identity requirement; it does not establish that the grading implementation was complete before inference. The amendment and its independent parity audit are included in the provenance inventory.

The independent audit confirms that exactly thirty outcomes change solely at the registration check: twenty-four primary and six control Astra records. The initial registry-artifact totals of 39/72 primary and 6/12 controls become 63/72 and 12/12, respectively. GPT-5.4 and Sol grades do not change. The amended grader preserves every non-identity check, rejects wrong or unknown identifiers, and produces no changed grades when replayed on 1,260 historical attempts. All thirty-six narrative judgments predate the diagnosis and remain fixed. These checks establish the amendment's narrow scope, not an error-free pre-run grader audit.

All 252 application outcomes are retained, with 249 HTTP 200 and three HTTP 422 responses. There are no infrastructure interruptions or missing request times in this campaign; the prespecified complete-triplet sensitivity excludes no attempts and the unknown-outcome bounds collapse to the observed scores. Initial provider-call inputs are byte-equivalent modulo the selected model across all 81 inference-bearing triplets. The 282 scored provider calls and six non-scored preflight calls have separate attribution. Subsequent repair or narration may depend on each model's earlier output.

## 5.7 Four-model GPT comparison with prospective identity checks

The ninth campaign adds `gpt-5.5`, displayed as GPT-5.5, alongside fresh `gpt-5.4`, `gpt-6-astra`, and `gpt-6.1-sol` observations. It uses the same historical CERRA application and all 28 frozen task records, including the 24 primary tasks, four controls, source snapshots, independent oracles, prompts, normal application repair, scope reconciliation, and browser contracts. Only the additional model routing/catalog entries and isolated local service ports change in the application fixture. The known MD01 scope override and MD04 annual-row ambiguity remain in scope for diagnostic attribution. This is a new contemporaneous comparison; it does not replace the earlier GPT results or execute the proposed English v0.2 benchmark.

All four models explicitly request medium reasoning effort, an 8192-token combined reasoning/visible-output cap, a 60-second model-call timeout, disabled storage, no temperature parameter, and the same JSON-object planner instruction through the regional US Responses endpoint. The frozen expected response identifiers are `gpt-5.4-2026-03-05`, `gpt-5.5-2026-04-23`, `gpt-6-astra`, and `gpt-6.1-sol`, respectively. The first two are

dated provider identifiers; the latter two remain aliases. None supplies an independently retained weight snapshot. Matching the exposed request settings does not identify or equalize realized provider compute, GPU type, memory allocation, energy, or monetary cost.

**Identity checks before collection.** The canonical HIN grader includes the exact manifest-authorized GPT-5.5 identifier and the already disclosed Astra registration before this campaign is frozen. The same canonical grader is used in preflight and scoring. The protocol wrapper also requires a requested application identity in the four-model manifest and an exact returned-identity match. Positive fixtures and wrong/unknown-identity negatives cover HIN deterministic and narrative outcomes, Maryland API outcomes, and Maryland browser outcomes. These are prospective controls for the ninth campaign. The eighth campaign's post-run Astra registration amendment remains a separate historical audit event and is not relabeled as a pre-run correction. Historical base-grader replay uses the appropriate published canonical grades rather than retrospectively applying the new four-model eligibility list to older model configurations.

**Adjacent blocks and order balance.** The schedule contains 336 application attempts: 288 primary and 48 controls, comprising 108 browser and 228 API attempts. Each task–repetition forms an adjacent four-model block. Manifest-order indices are 0=GPT-5.4, 1=GPT-5.5, 2=Astra, and 3=Sol. The Williams rows are $[0,1,3,2]$, $[1,2,0,3]$, $[2,3,1,0]$, and $[3,0,2,1]$. A single Python `Random(20261002)` stream shuffles primary tasks and controls separately, assigning six primary tasks and one control to each base row. Repetition $r = 0,1,2$ uses row $(\text{base} + r) \bmod 4$; a separate `Random(20261002+r)` stream shuffles the 28 task IDs within each repetition.

Each model occupies every position six times among primary tasks and once among controls per repetition; the corresponding three-repetition counts are 18 and three. Each directed immediate predecessor occurs 18 times in primary blocks and three times in control blocks. Each model pair is earlier/later 36/36 times among primary tasks and 42/42 overall. Only four of the 24 possible full permutations occur, 21 times each. Each model visits three distinct positions for a given task and omits the fourth, so complete per-task position balance is not claimed. This design balances exposure within adjacent task blocks, not transitions between different tasks or time-varying provider load. The twelve MD16 controls terminate in the shared deterministic precheck and supply no model-inference evidence.

**Prespecified missingness and analysis.** Started attempts without retained application outcomes are settled once as infrastructure non-successes, never reconstructed from provider output and never retried. Resumption, if needed, starts only unscheduled attempts. All scheduled primary attempts remain in the primary score, while unknown latency is omitted without imputation and explicitly counted. A prespecified complete-block sensitivity removes any affected task–repetition block from all four configurations, preserving unequal remaining task-cluster sizes with attempt-weighted resampling; unknown-outcome bounds are separate. This does not replace the equally task-weighted primary analysis or attribute infrastructure loss to model quality.

The primary score is the mean of 24 task-level three-repetition success means. The frozen analysis specifies 10,000 shared task-cluster bootstrap draws with seed 20261002 and linearly interpolated percentile 95% intervals. All six within-campaign pairwise score contrasts are exploratory and unadjusted for multiple comparisons. Marginal uncertainty for median application-request latency is also specified before inference: each sampled task contributes all recorded repetition timings, including failures; the pooled median is recomputed per draw and unknown durations are omitted. This differs from the explicitly post-hoc median intervals introduced for the earlier eighteen observations on 1 October. Separate horizontal success and vertical median-latency intervals do not form a joint 95% ellipse, a P90 range, or a paired latency-difference test. They remain conditional on this task suite and the recorded outcomes.

The protocol requires 48 identity-blind narrative judgments against the unchanged rubric, instructions, and historical calibration, with packet and decision hashes sealed before aggregate release. This procedure is AI-assisted rather than human domain-expert validation. The blinded reviewer must not receive model mappings, trial/provider logs, model names or dated identity suffixes, or current aggregate results. Completion and audit counts are reported with the final results rather than inferred from the scheduled totals.

# 6 Results

## 6.1 Workflow success across campaigns (RQ1)

Table 1 reports twenty-two configuration–campaign observations across nine campaigns. A model identifier may recur with different settings; these observations are not thirteen immutable runtime configurations. Strict success ranges from 34/72 to 66/72 across the historical production configurations, but the broad intervals and environment differences preclude an intrinsic ranking. For example, Qwen3 8B passes 51/72 overall yet succeeds on only 6/15 supported filter attempts, illustrating why aggregate correctness cannot substitute for an interaction-specific result.

In the September 24 historical campaign, Gemma 4 12B records 58/72 strict successes, 15/15 supported filter successes, and no observed UI-state violations in its 21 primary browser attempts. Its paired advantage over Qwen3 14B in the September 24 campaign is 9.7 points, with interval [−8.3, 29.2]. GPT-5.4 records 66/72 primary successes, but its 11.1-point advantage over GPT-5.4 mini has interval [−4.2, 26.4]. Neither interval excludes zero; descriptive ordering should not be presented as a conclusive superiority claim.

Across all nine campaigns, recorded application outcomes comprise 1,795 status-200 responses, 52 status-422 responses, and one interrupted outcome without a retained application response. Successful HTTP delivery does not imply strict task success. A completed request can select the wrong scope, omit a required qualification, or change state incorrectly.

## 6.2 Success and response time in one view (RQ2)

Figure 2 places all twenty-two configuration–campaign observations in the same coordinate system, with greater strict success to the right and shorter median request latency below. Repeated identifiers remain separate points because their runtime and experimental conditions differ. Campaign labels identify this context, and the latest four-model comparison is enlarged for readability. The labels refer to the same observations as Table 1; Table 2 makes requested reasoning and token limits explicit.

The joint display exposes the observed Qwen3.8 trade-off without conflating its longer response time with a lower score. In the latest matched campaign, the GPT-5.4/GPT-5.5/Astra/Sol medians are 3.91/3.35/2.54/3.24 seconds, with marginal 95% median intervals [2.19, 5.61], [2.08, 4.95], [2.07, 3.47], and [2.90, 4.13] seconds. GPT-5.5 has the highest observed strict-success point and Astra the shortest observed median. Separate marginal intervals do not establish paired latency differences; the six exploratory paired score contrasts remain in Table 3. The historical points are context rather than a controlled leaderboard.

## 6.3 Four-model GPT results with GPT-5.5 (RQ2)

The ninth campaign produces fresh observations under the same requested medium reasoning and 8192-token combined budget: GPT-5.4 63/72 (87.5%; interval [75.0%, 100.0%]); GPT-5.5 68/72 (94.4%; interval [84.7%, 100.0%]); GPT-6 Astra 63/72 (87.5%; interval [75.0%, 100.0%]); GPT-6.1 Sol 65/72 (90.3%; interval [77.8%, 100.0%]). Each interval is a marginal 95% task-cluster interval. GPT-5.5 has the highest observed score in this campaign. Its GPT-5.5-minus-GPT-5.4 difference is +6.94 points, with paired interval [−2.78, +19.44]. All six contrasts in Table 3 include or touch zero; they establish neither a stable model ranking nor equivalence. The new observations do not replace the earlier GPT scores.

Primary request median/P90 timing is GPT-5.4 3.91/11.35 seconds (median interval [2.19, 5.61] seconds); GPT-5.5 3.35/9.02 seconds (median interval [2.08, 4.95] seconds); GPT-6 Astra 2.54/5.06 seconds (median interval [2.07, 3.47] seconds); GPT-6.1 Sol 3.24/6.81 seconds (median interval [2.90, 4.13] seconds). All four timing denominators are 72; measured failures remain included. Figure 2 displays these prospectively specified median intervals as vertical whiskers beside the unchanged historical points. Astra has the shortest observed median in the new run. The intervals are marginal, not estimates of pairwise latency differences, and do not measure provider-internal compute. Median primary-browser completion is 7.12, 7.03, 7.07, and 8.30 seconds for GPT-5.4, GPT-5.5, Astra, and Sol, respectively. All strict successes finish within the 30-second completion budget.

Every configuration passes 12/12 separate controls and 15/15 supported filter changes, with zero observed UI-state violations in 21 primary browser attempts. Controls remain outside the primary comparison. Twenty-one tasks pass all three repetitions for GPT-5.4, Astra, and Sol; twenty-two do so for GPT-5.5. The functional-equivalence sensitivity adds no successes to these strict totals. Maryland-query scores are 18/27, 24/27, 18/27, and 20/27 in manifest order. HIN primary scores are 24/24, 23/24, 24/24, and 24/24. These task-family counts are diagnostics of the same small suite, not independent domain-competency estimates.

All 336 application outcomes and timings are retained: 334 HTTP 200 and two GPT-5.4 HTTP 422 validation rejections on MD07. There are no infrastructure interruptions, so the prespecified complete-block sensitivity excludes nothing and unknown-outcome bounds collapse to the observed values. Initial provider inputs match across all 81 inference-bearing four-model blocks. The 376 scored provider calls and eight non-scored preflight calls are separately attributed; all 384 provider calls return HTTP 200, which does not make every resulting application answer successful. GPT-5.5 and Astra were registered in the canonical grader before freezing; no post-run identity correction occurs in this ninth campaign.

**Task-specific differences behind the point estimate.** All twelve MD01 raw plans correctly select filtered scope. The unchanged application overrides that choice to viewport scope and returns zero rather than the oracle's 266 reports. This is a shared downstream defect, not twelve independently demonstrated planning errors. On MD04, GPT-5.5 supplies the correct total of 151,679 and all eight annual rows in all three attempts;

Table 1: 22 observations from 13 CERRA model identifiers across 9 campaigns. Strict and functional scores use 72 primary attempts per observation; controls are excluded. Intervals are 95% task-cluster bootstrap intervals. Request latency retains failed primary attempts with recorded timing. Runtime paths and dates differ between campaigns. Requested reasoning appears beneath each model; Table 2 details token limits.

| Configuration | Strict success | 95% CI | Median (s) | P90 (s) | UI actions | UI-state violations | Functional |
|---|---|---|---|---|---|---|---|
| *September 18, 2026 — Production site* | | | | | | | |
| A1 Qwen3 8B<br>Thinking off | 51/72 (70.8%) | 52.8–87.5 | 1.06 | 2.17 | 6/15 | 6/21 | 51/72 |
| A2 Llama 3.1 8B<br>Not set (local) | 40/72 (55.6%) | 36.1–75.0 | 2.00 | 3.77 | 14/15 | 7/21 | 40/72 |
| A3 Gemma 3 4B<br>Not set (local) | 34/72 (47.2%) | 27.8–66.7 | 2.40 | 4.18 | 12/15 | 3/21 | 48/72 |
| *September 23, 2026 — Production site* | | | | | | | |
| B1 GPT-4.1 mini<br>Not explicitly set | 57/72 (79.2%) | 62.5–93.1 | 1.20 | 2.59 | 12/15 | 3/21 | 63/72 |
| B2 GPT-5.4 mini<br>Not explicitly set | 58/72 (80.6%) | 65.3–93.1 | 1.12 | 2.76 | 14/15 | 0/21 | 62/72 |
| B3 GPT-5.4<br>Not explicitly set | 66/72 (91.7%) | 79.2–100.0 | 1.27 | 2.94 | 12/15 | 0/21 | 66/72 |
| *September 24, 2026 — Production site* | | | | | | | |
| C1 Qwen3 14B<br>Thinking off | 51/72 (70.8%) | 54.2–87.5 | 2.93 | 5.56 | 12/15 | 3/21 | 57/72 |
| C2 Gemma 4 12B<br>Thinking off | 58/72 (80.6%) | 63.9–95.8 | 3.51 | 6.71 | 15/15 | 0/21 | 61/72 |
| *September 29, 2026 — Paired local* | | | | | | | |
| D1 Qwen3 14B<br>Thinking off | 49/72 (68.1%) | 50.0–84.7 | 3.77 | 7.20 | 11/15 | 3/21 | 55/72 |
| D2 Qwen3.8 27B Q4_K_M<br>Thinking off | 63/72 (87.5%) | 75.0–100.0 | 14.50 | 29.81 | 15/15 | 0/21 | 63/72 |
| *September 29 API extension — Frozen local app / API* | | | | | | | |
| E1 GPT-6.1 Sol<br>Medium | 64/72 (88.9%) | 76.4–100.0 | 2.15 | 4.88 | 15/15 | 0/21 | 64/72 |
| *September 29 lighter-model comparison — Paired local / lighter model* | | | | | | | |
| F1 Gemma 4 12B<br>Thinking off | 60/72 (83.3%) | 66.7–95.8 | 3.18 | 6.35 | 15/15 | 0/21 | 63/72 |
| F2 Qwen3.5 9B Q4_K_M<br>Thinking off | 59/72 (81.9%) | 66.7–95.8 | 2.95 | 6.19 | 12/15 | 0/21 | 62/72 |
| *September 30 matched API comparison — Matched local app / API* | | | | | | | |
| G1 GPT-5.4<br>Medium | 65/72 (90.3%) | 77.8–100.0 | 3.51 | 8.58 | 15/15 | 0/21 | 65/72 |
| G2 GPT-6.1 Sol†<br>Medium | 63/72 (87.5%) | 73.6–98.6 | 2.83 | 5.97 | 15/15 | 0/21 | 63/72 |
| *September 30 three-model API comparison — Three-model matched local app / API* | | | | | | | |
| H1 GPT-5.4<br>Medium | 64/72 (88.9%) | 76.4–98.6 | 3.82 | 11.16 | 14/15 | 0/21 | 65/72 |
| H2 GPT-6 Astra<br>Medium | 63/72 (87.5%) | 75.0–100.0 | 2.49 | 4.92 | 15/15 | 0/21 | 63/72 |
| H3 GPT-6.1 Sol<br>Medium | 64/72 (88.9%) | 75.0–100.0 | 3.19 | 5.91 | 15/15 | 0/21 | 64/72 |
| *October 2 four-model API comparison — Four-model matched local app / API* | | | | | | | |
| I1 GPT-5.4<br>Medium | 63/72 (87.5%) | 75.0–100.0 | 3.91 | 11.35 | 15/15 | 0/21 | 63/72 |
| I2 GPT-5.5<br>Medium | 68/72 (94.4%) | 84.7–100.0 | 3.35 | 9.02 | 15/15 | 0/21 | 68/72 |
| I3 GPT-6 Astra<br>Medium | 63/72 (87.5%) | 75.0–100.0 | 2.54 | 5.06 | 15/15 | 0/21 | 63/72 |
| I4 GPT-6.1 Sol<br>Medium | 65/72 (90.3%) | 77.8–100.0 | 3.24 | 6.81 | 15/15 | 0/21 | 65/72 |

UI actions: supported filter changes. UI-state violations: actual value changes violating the frozen task policy among 21 primary browser attempts; redundant unchanged fields are excluded. Functional equivalence was a post-hoc diagnostic in the initial campaign, inherited by later campaigns; it does not replace strict success. The September 29 paired-local baseline is the contemporaneous Qwen3 14B rerun. Campaigns sharing a date remain separate: the API extension, original Qwen pair, and later lighter-model pair are not pooled. Gemma 4 12B is rerun in the lighter-model campaign. † September 30 matched API comparison, GPT-6.1 Sol: 1 interrupted application outcome remains primary non-success; request timing is available for 71/72 attempts. Unknown completion time is excluded, not imputed.

Table 2: Requested reasoning and token limits for every configuration–campaign observation in Table 1 and Figure 2. Labels describe the request, not measured internal compute. Local thinking switches do not map to an ordinal API reasoning-effort scale.

| ID | Configuration | Requested reasoning | Output budget per model call | Local context |
|---|---|---|---|---|
| *Sep 18 — production local* | | | | |
| A1 | Qwen3 8B | Thinking off | ≤1,000 generated | 12,000* |
| A2 | Llama 3.1 8B | Not set (local) | ≤1,000 generated | 16,384 |
| A3 | Gemma 3 4B | Not set (local) | ≤1,000 generated | 16,384 |
| *Sep 23 — production API* | | | | |
| B1 | GPT-4.1 mini | Not explicitly set | 300/350 planner; 700 narration | — |
| B2 | GPT-5.4 mini | Not explicitly set | 300/350 planner; 700 narration | — |
| B3 | GPT-5.4 | Not explicitly set | 300/350 planner; 700 narration | — |
| *Sep 24 — production local* | | | | |
| C1 | Qwen3 14B | Thinking off | ≤1,000 generated | 8,192 |
| C2 | Gemma 4 12B | Thinking off | ≤1,000 generated | 8,192 |
| *Sep 29 — paired local (Qwen)* | | | | |
| D1 | Qwen3 14B | Thinking off | ≤1,000 generated | 8,192 |
| D2 | Qwen3.8 27B | Thinking off | ≤1,000 generated | 8,192 |
| *Sep 29 — Sol API extension* | | | | |
| E1 | GPT-6.1 Sol | Medium | 8,192 combined reasoning + visible | — |
| *Sep 29 — paired local (lighter)* | | | | |
| F1 | Gemma 4 12B | Thinking off | ≤1,000 generated | 8,192 |
| F2 | Qwen3.5 9B | Thinking off | ≤1,000 generated | 8,192 |
| *Sep 30 — matched API pair* | | | | |
| G1 | GPT-5.4 | Medium | 8,192 combined reasoning + visible | — |
| G2 | GPT-6.1 Sol | Medium | 8,192 combined reasoning + visible | — |
| *Sep 30 — matched API triplet* | | | | |
| H1 | GPT-5.4 | Medium | 8,192 combined reasoning + visible | — |
| H2 | GPT-6 Astra | Medium | 8,192 combined reasoning + visible | — |
| H3 | GPT-6.1 Sol | Medium | 8,192 combined reasoning + visible | — |
| *Oct 2 — matched API quartet* | | | | |
| I1 | GPT-5.4 | Medium | 8,192 combined reasoning + visible | — |
| I2 | GPT-5.5 | Medium | 8,192 combined reasoning + visible | — |
| I3 | GPT-6 Astra | Medium | 8,192 combined reasoning + visible | — |
| I4 | GPT-6.1 Sol | Medium | 8,192 combined reasoning + visible | — |

Medium is an explicitly requested API effort. Not explicitly set means the historical request omitted that parameter; provider defaults and realized effort were not retained and are not reconstructed. Not set (local) means no thinking control was sent; it is not a claim about unsupported model capabilities. Thinking off records an explicit false switch. *Qwen3 8B uses a 12,000-token input limit in the retained Transformers service, not an Ollama context setting. Local 8,192/16,384 context settings limit context rather than output; the bridge separately clamps requested generation to at most 1,000 tokens. Actual planner/narrator requests can be smaller. The API 8,192 cap includes reasoning plus visible output. None of these caps guarantees equal realized compute; tokenizers, hardware, and other settings differ. All ten later GPT observations request medium; no retained campaign explicitly requests ultra.

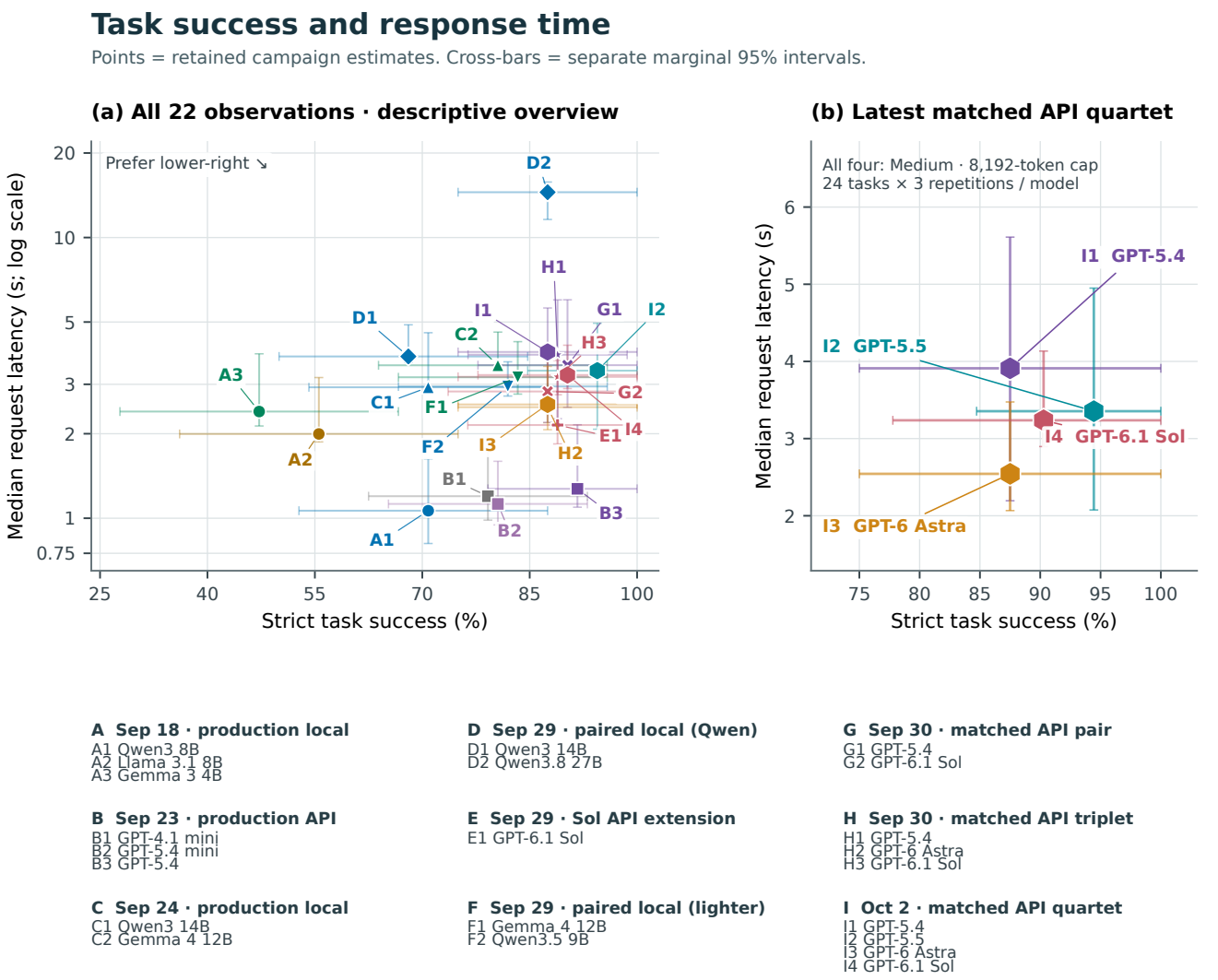



Figure 2: Strict task success (horizontal) and median application-request latency (vertical), retaining all twenty-two configuration–campaign observations. Each point corresponds to 72 primary scheduled attempts; measured timing includes failures, with 71 observed times for the earlier interrupted Sol observation and 72 for every other point. Horizontal whiskers show 95% task-cluster score intervals. Vertical whiskers show marginal 95% task-cluster intervals for the median: the historical eighteen retain the post-hoc analysis (10,000 draws, seed 20261001), whereas the latest four use the prospective ninth-campaign protocol (10,000 draws, seed 20261002). These are separate marginal intervals, not a joint confidence ellipse, a P90 range, or a paired-difference test. The overview uses a logarithmic latency axis to keep both short and long timings visible; the latest-GPT panel uses a linear axis. Different historical settings, dates, and execution paths prevent a controlled global ranking. The preferred direction is rightward and downward, conditional on those conditions.

Table 3: All six ninth-campaign score contrasts, first model minus second, in percentage points. GPT-5.4/5.5 are abbreviated 5.4/5.5. Intervals are paired 95% task-cluster percentile intervals, exploratory and unadjusted for multiplicity.

| Contrast | Difference | 95% interval |
|---|---|---|
| 5.4 − 5.5 | −6.94 | [−19.44, +2.78] |
| 5.4 − Astra | +0.00 | [−12.50, +12.50] |
| 5.4 − Sol | −2.78 | [−15.28, +9.72] |
| 5.5 − Astra | +6.94 | [−2.78, +20.83] |
| 5.5 − Sol | +4.17 | [−2.78, +13.89] |
| Astra − Sol | −2.78 | [−8.33, +0.00] |

Sol supplies the required rows twice, Astra gives the correct aggregate without yearly rows, and GPT-5.4 refuses all three. The frozen rubric requires annual rows although the prompt does not explicitly request them, so this contrast partly concerns an ambiguous output contract.

GPT-5.5, Astra, and Sol pass all three selected-crash MD07 queries. GPT-5.4 fails two through validation rejection and one through an unsupported-operation response. Conversely, GPT-5.4 and GPT-5.5 pass all three supported MD09 original-versus-estimated location comparisons, while Astra and Sol refuse them. GPT-5.5's higher point estimate therefore combines different task-level behaviors; it does not imply superiority across every capability.

The sealed identity-blind review accepts 47 of 48 narratives. One GPT-5.5 HIN14 answer correctly refuses to estimate a causal road-diet effect but additionally asserts that project designs and exposure changes are absent. The supplied evidence does not establish those extra absences, so the reviewer fails claim support for that response. The causal refusal itself is correct; the unsupported additional claims cause the failure. This is one independent AI review under the frozen rubric, not human domain-expert validation or evidence of inter-rater agreement.

## 6.4 Historical single-configuration API observation

GPT-6.1 Sol passes 64/72 primary attempts (88.9%), with a 95% task-cluster interval of [76.4%, 100%]. Twenty-one of the 24 tasks pass all three repetitions. All 15 supported filter attempts and all 21 primary browser attempts pass, with zero observed UI-state violations. The twelve controls and twelve identity-blind narrative assessments also pass; controls remain outside the primary score. The inherited functional-equivalence sensitivity remains 64/72, so it adds no passes to this configuration's strict score.

Median request latency is 2.15 seconds and P90 is 4.88 seconds; median primary-browser completion is 7.04 seconds. All 64 strict successes finish within the 30-second budget. These measurements describe the frozen local-application/API route in Section 5.3, not a matched speed comparison with other campaigns. All 84 requests return HTTP 200, but eight primary attempts fail strict scoring: three on MD01, three on MD04, and two on MD09. This contrast isolates the difference between successful request delivery and successful analytical assistance. Neither the score nor the latency establishes superiority over the historical GPT configurations or local models without a contemporaneous comparison.

## 6.5 Matched GPT settings and interruption sensitivity (RQ2)

Figure 3 reports the fresh API comparison. GPT-5.4 records 65/72 strict successes (90.3%; marginal 95% task-cluster interval [77.8%, 100%]); GPT-6.1 Sol records 63/72 (87.5%; [73.6%, 98.6%]), with the interrupted attempt counted as a non-success. The paired Sol-minus-GPT difference is −2.8 points, with interval [−12.5, 4.2]. This interval establishes neither superiority nor equivalence on the tested workflow. Historical GPT-5.4 and Sol observations remain 66/72 and 64/72 under their earlier campaign conditions; the matched results do not overwrite or pair against them.

Both configurations pass all 15 supported filter attempts, all 21 primary browser attempts, and all twelve controls, with zero observed UI-state violations. Both pass all 24 HIN primary attempts and all twelve reviewed narratives; Maryland-query scores are 20/27 for GPT-5.4 and 18/27 for Sol. Twenty-one tasks pass all three repetitions for GPT-5.4 versus twenty for Sol. Functional-equivalence sensitivity adds no passes to either configuration. These are CERRA task outcomes, not general domain competency estimates.

Median/P90 request latency is 3.51/8.58 seconds for GPT-5.4 over 72 measured primary requests, and 2.83/5.97 seconds for Sol over 71. The unknown Sol completion time is excluded; the missing observation prevents an unconditional claim about all scheduled latency outcomes. Median primary-browser completion is 7.18 versus 7.38 seconds. Every strict success completes within the 30-second budget. Matched request settings do not identify provider-internal compute or establish a token-throughput comparison, and the added marginal median intervals in Figure 2 do not estimate a paired latency effect.

The post-interruption complete-pair sensitivity excludes both repetition-2 MD07 observations: Sol records 63/71 and GPT-5.4 64/71. Its attempt-weighted Sol-minus-GPT difference is −1.41 points, with paired task-cluster interval [−11.27, 5.80]. This is a secondary estimand with one smaller task cluster, not a replacement primary score. Assigning the single unknown Sol outcome success or failure bounds its full-denominator score at 63–64/72 and the paired point difference at

−2.78 to −1.39 points. That arithmetic bound does not resolve the unknown outcome or constitute a confidence interval. The primary, complete-pair sensitivity, and bound are reported together to make the infrastructure event visible.

## 6.6 Three-model matched API results and registration audit (RQ2)

Figure 4 and Table 4 report the eighth campaign separately from both earlier GPT campaigns. GPT-5.4 records 64/72 strict successes (88.9%; marginal 95% task-cluster interval [76.4%, 98.6%]); GPT-6 Astra records 63/72 strict successes (87.5%; marginal 95% task-cluster interval [75.0%, 100.0%]); GPT-6.1 Sol records 64/72 strict successes (88.9%; marginal 95% task-cluster interval [75.0%, 100.0%]). These final scores use the post-run model-registration amendment described in Section 5.6; the initial HIN identity-allowlist rejections are preserved as audit records, not interpreted as task-performance evidence.

All three pairwise differences and intervals are reported with their directions explicitly labeled. They are exploratory and unadjusted for multiple comparisons, and all include or touch zero (the Astra-minus-Sol upper endpoint is exactly zero). A descriptive score ordering does not establish a universal winner, and an interval containing zero does not establish equivalence. This campaign evaluates the same CERRA tasks, not the proposed English v0.2 core.

Primary request median/P90 latency is GPT-5.4: 3.82/11.16 seconds over 72/72 measured primary requests; GPT-6 Astra: 2.49/4.92 seconds over 72/72 measured primary requests; GPT-6.1 Sol: 3.19/5.91 seconds over 72/72 measured primary requests. Measured failures remain in these distributions; unmeasured completion times are never imputed. New marginal median-latency intervals are reported in Figure 2; no paired latency-difference interval is estimated. Provider-internal GPU hardware, memory use, and realized compute are unknown, so these application timings are not a token-throughput or equal-compute comparison.

Median primary-browser completion times are GPT-5.4 7.38 seconds, GPT-6 Astra 7.20 seconds, GPT-6.1 Sol 8.10 seconds. All strict successes finish within the 30-second completion budget. Supported filter successes are GPT-5.4 14/15, GPT-6 Astra 15/15, GPT-6.1 Sol 15/15; observed UI-state violations are GPT-5.4 0/21, GPT-6 Astra 0/21, GPT-6.1 Sol 0/21. The separate controls score GPT-5.4 12/12, GPT-6 Astra 12/12, GPT-6.1 Sol 12/12. Controls remain outside the primary score and paired inference; the deterministic MD16 control supplies no model-quality evidence.

Table 4: All three contemporaneous score contrasts in the eighth campaign. Differences and 95% paired task-cluster intervals are percentage points, with the first model minus the second. These exploratory intervals are unadjusted for multiple comparisons.

| Contrast | Difference | 95% interval |
|---|---|---|
| GPT-5.4 − Astra | +1.39 | [−11.11, +15.28] |
| GPT-5.4 − Sol | +0.00 | [−12.50, +12.50] |
| Astra − Sol | −1.39 | [−4.17, +0.00] |

## 6.7 Paired quality–latency trade-off (RQ2)

Figure 5 presents the original Qwen3 14B–Qwen3.8 27B pair. Qwen3.8 27B passes 63/72 primary attempts (87.5%) versus 49/72 (68.1%) for Qwen3 14B. The paired difference is 19.4 points, with interval [1.4, 38.9]. The observed fixed-suite difference favors the newer, larger configuration, and the interval remains above zero under empirical task-cluster resampling. This describes sensitivity to the tested task mix, not a sampled deployment population or an isolated model-generation effect.

The number of tasks passing every repetition increases from 15/24 to 21/24. Supported filter success rises from 11/15 to 15/15. UI-state violations fall from 3/21 to 0/21 primary browser attempts; all three baseline violations are premature partial actions on the ambiguous Baltimore task discussed below. Zero observed UI-state violations on seven tasks is not evidence of universal safe automation.

This improvement comes with longer response times. Median request latency increases from 3.77 to 14.50 seconds, and P90 from 7.20 to 29.81 seconds. Median browser completion increases from 8.70 to 15.50 seconds. With a 30-second completion budget, strict successes are 49/72 for the baseline and 55/72 for Qwen3.8, compared with 49/72 and 63/72 without that budget. Thus eight of the newer configuration's strict successes finish after the deadline. This is a descriptive view of service usefulness, not an independently validated user-tolerance threshold.

Both models produce 69/72 status-200 primary responses and three 422 failures, so HTTP completion alone conceals their success difference. Controls score 9/12 and 12/12, respectively, and are excluded from the paired inference.

## 6.8 Differences by task family

For the original Qwen3 14B–Qwen3.8 27B comparison, both configurations pass all 15 HIN source-query attempts. HIN reasoning and limitation checks improve from 8/9 to 9/9. Maryland data queries improve from 15/27 to 18/27, while boundary tasks improve from 0/6 to 6/6. Supported filter actions improve from 11/15

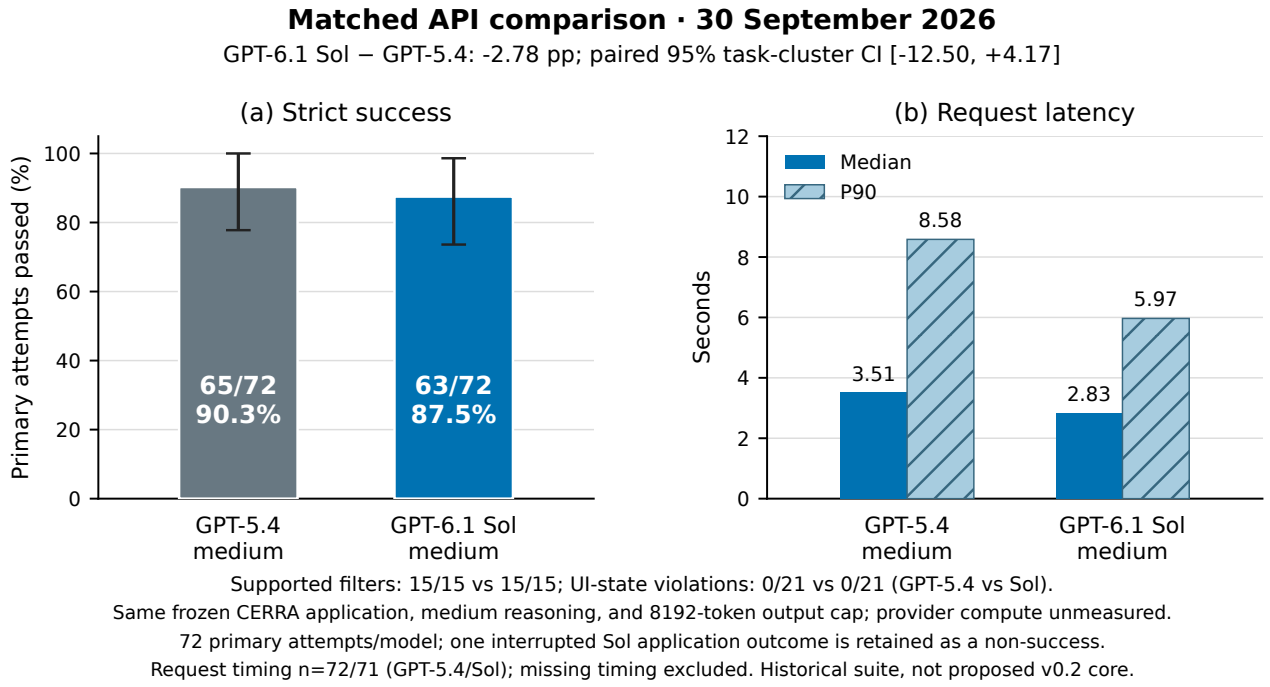



Figure 3: Matched GPT API follow-up on the original CERRA task suite. Score whiskers are marginal 95% task-cluster intervals; the paired Sol-minus-GPT effect is −2.8 points [−12.5, 4.2]. One interrupted Sol application outcome counts as a primary non-success; median/P90 timing uses 72 GPT-5.4 and 71 Sol observations. Matching medium reasoning and the 8192-token cap does not establish equal realized compute. Section 6.5 reports the complete-pair and one-attempt-bound sensitivities.

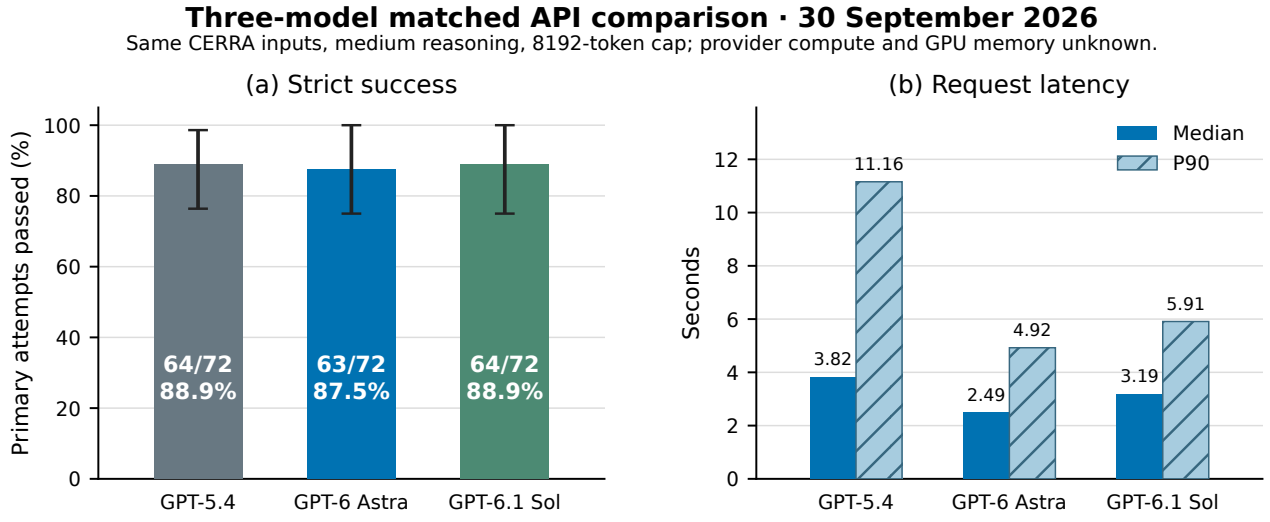



Figure 4: Eighth-campaign comparison of GPT-5.4, GPT-6 Astra, and GPT-6.1 Sol under matched CERRA inputs, medium reasoning, and an 8192-token output cap. Score whiskers are marginal 95% task-cluster intervals; Table 4 reports all three exploratory pairwise intervals without multiplicity adjustment. Scores use the disclosed post-run identity-registration amendment. Request timing denominators are shown explicitly; provider compute and GPU memory are unobserved.

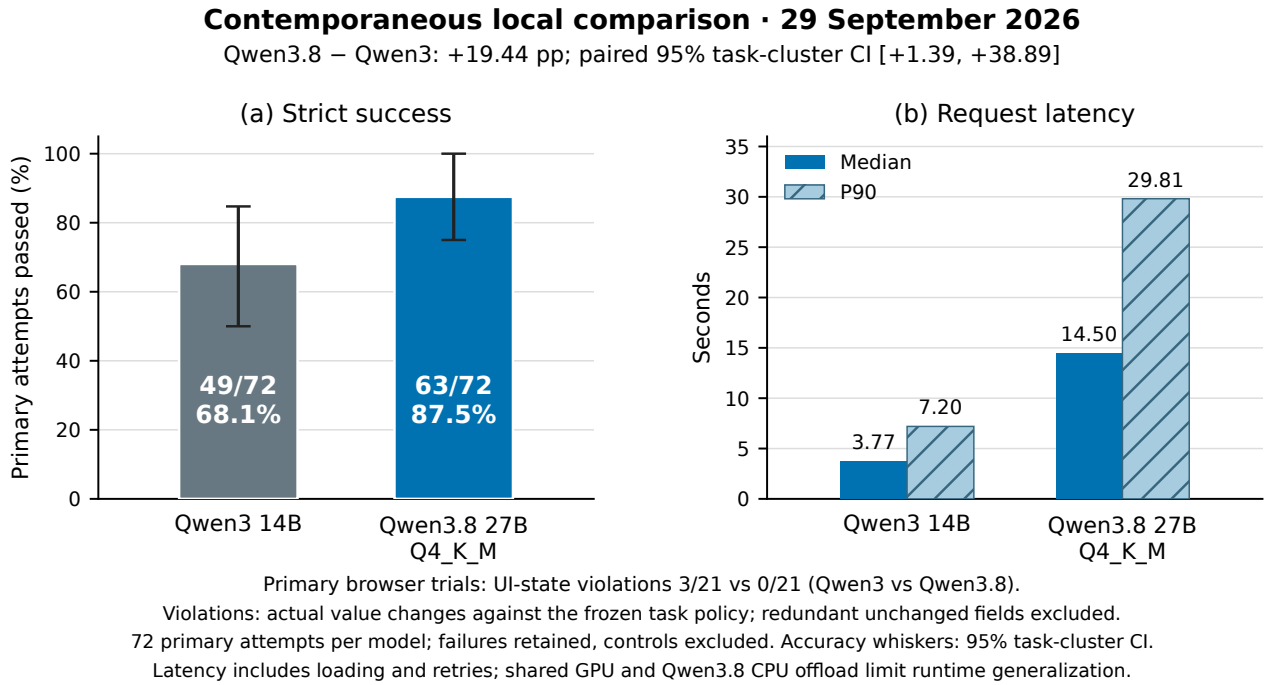



Figure 5: September 29 contemporaneous local comparison. Left: strict primary success with 95% task-cluster bootstrap intervals. Right: median and P90 request latency over all 72 primary attempts per configuration. The paired success difference is 19.4 points [1.4, 38.9]. Historical production latency is not included in this figure.

to 15/15. The improvement is therefore distributed unevenly and does not eliminate data-query failures.

These category counts are diagnostic subdivisions of the same small suite. They are not independent experiments, and we do not assign category-level significance claims. Repeated failure on a single task contributes three failures, emphasizing the importance of reviewing the task-level records.

### 6.9 A lighter-model trade-off without established equivalence (RQ2)

Figure 6 shows the later local pair. Qwen3.5 9B passes 59/72 primary attempts (81.9%), versus 60/72 (83.3%) for Gemma 4 12B. Both marginal 95% task-cluster intervals are [66.7%, 95.8%]. The paired Qwen-minus-Gemma difference is −1.4 points, with interval [−20.8, 19.4]. This wide interval establishes neither superiority nor equivalence; a nonsignificant difference is not evidence that the configurations are interchangeable. Nineteen tasks pass all three repetitions for Qwen and twenty for Gemma. Both pass all twelve controls, which remain excluded from primary inference.

Median/P90 request latency is 2.95/6.19 seconds for Qwen and 3.18/6.35 seconds for Gemma. The observed median difference is 0.23 seconds, without a separate paired latency uncertainty estimate. Median primary-browser completion is 5.39 and 7.59 seconds, respectively; task failures can omit an intended UI action, so shorter completion alone does not imply better interaction. All strict successes meet the 30-second completion budget. Qwen returns HTTP 200 for all 72 primary attempts, whereas Gemma returns 66 status-200 responses and six 422 failures. Delivery success again differs from task success.

Both configurations have zero UI-state violations in 21 primary browser attempts, but supported filter success is 12/15 for Qwen versus 15/15 for Gemma. Qwen leaves the requested GPS cross-check filter unchanged on all three RV02 attempts. Preserving state is therefore insufficient when a supported action was explicitly requested. Both pass all 15 HIN source-query attempts. HIN reasoning scores are 8/9 for Qwen and 9/9 for Gemma; Maryland query scores are 21/27 and 18/27; both score 3/6 on UI boundaries. Functional-equivalence sensitivity yields 62/72 and 63/72, respectively, without replacing the primary scores or their estimand. These category diagnostics reuse the same small task suite and supply no independent significance tests.

## 7 Failure Analysis (RQ3)

### 7.1 Actual state violations versus strict patch failures

In the original Qwen3 14B–Qwen3.8 27B pair, MD17 asks to focus the map on “Baltimore” for 2020, with-

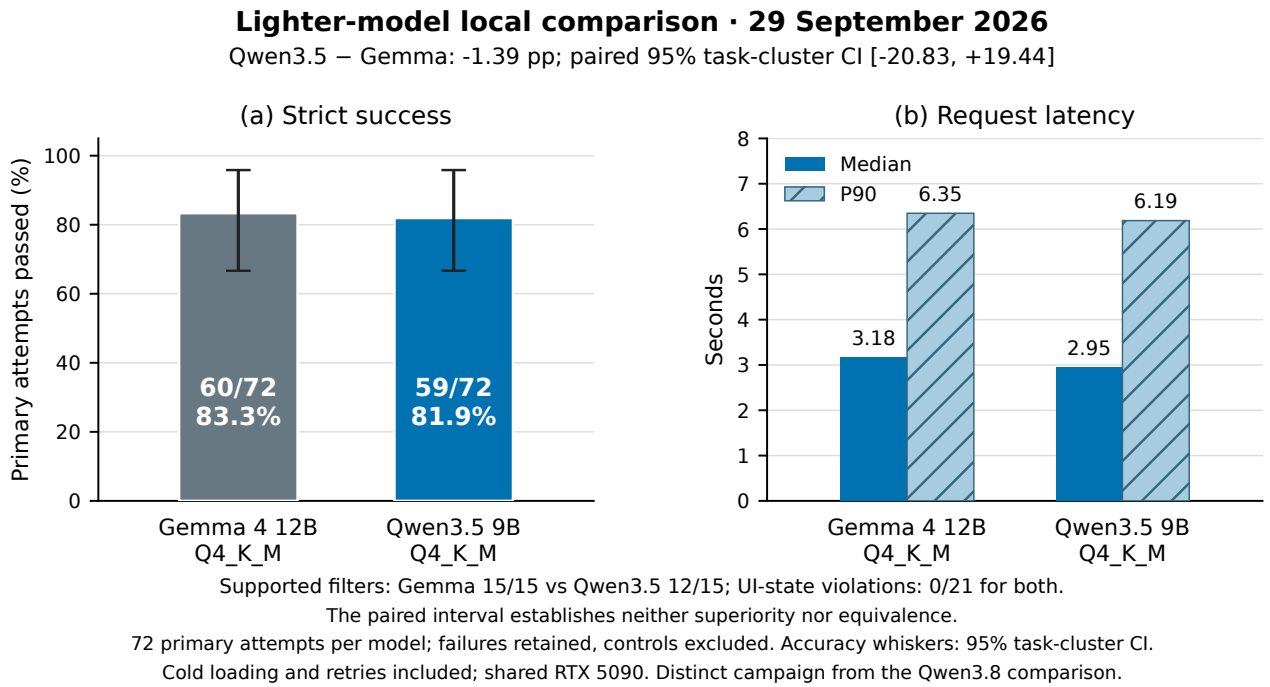



Figure 6: Separate lighter-model campaign: a fresh Gemma 4 12B comparison with Qwen3.5 9B. Strict-score whiskers are marginal 95% task-cluster intervals. The paired Qwen-minus-Gemma difference is −1.4 points [−20.8, 19.4]; this is not an equivalence result. Request timings include cold loading and failures under this campaign's shared-GPU conditions.

out specifying City or County. The Qwen3 14B baseline changes the year from 2024 to 2020 in all three attempts before resolving the geographical ambiguity. Qwen3.8 preserves state and requests clarification. The year change is explicitly requested, but applying it before geographical clarification violates the frozen preserve-state policy. These are premature partial actions under that policy; the experiment does not establish user harm or show that the year itself was unrequested.

By contrast, one baseline RV01 attempt includes redundant action fields whose values match existing controls. It fails the strict requirement to change only the requested fields, but produces no unwanted value change. Collapsing both phenomena into an "unsafe action" total would overstate the latter's observed effect. Strict contract adherence and actual mutation should remain separately inspectable.

On MD15, both configurations refuse an unsupported rain/fatal map filter and preserve state. The baseline nevertheless fails all three strict attempts because it omits the required explanation that weather can be queried but cannot be applied as that map filter. This is a missing capability qualification, not an unintended action.

## 7.2 Correct displayed values can still fail the contract

For MD05 in the original Qwen pair, both configurations display the correct count of 14,220 rain-associated crash reports in the requested 2022 statewide snapshot. The baseline chooses a disallowed breakdown operation instead of an accepted summary/weather operation, causing three strict failures. Describing this as numerical fabrication would be false. The inherited functional-equivalence sensitivity credits such narrowly defined cases without modifying the primary rubric.

Under that secondary analysis, the baseline improves from 49/72 to 55/72 and Qwen3.8 remains 63/72. The primary 19.4-point difference becomes an 11.1-point descriptive difference under the secondary interpretation. The primary paired interval should not be reused for this different estimand. The earlier Gemma 3 campaign shows an even larger sensitivity, moving from 34/72 to 48/72. These examples show that a structured interface contract is a consequential evaluation choice.

## 7.3 Task ambiguity and shared application constraints

MD04 asks for Baltimore County report totals for all years and all severity levels. Both configurations in the original Qwen pair return the correct aggregate, 151,679,

with appropriate filters. The frozen oracle expects separate annual rows, whereas the prompt does not explicitly require one row per year. Both fail all three strict attempts. The API extension repeats this pattern: the aggregate is correct in all three attempts, but the required annual rows are absent. We retain the scores and flag the ambiguity rather than retrospectively rewriting the oracle.

MD01 tests counting the complete current selection, including records outside the viewport. Every configuration–campaign observation fails all three attempts on this task. In the original Qwen-pair study, the baseline's final plan uses a selected-record path with zero evidence, while Qwen3.8 returns an unsupported outcome without the required evidence. In the API extension, all three final plans use viewport scope and display zero rather than the frozen selection's 266 reports.

The frozen application's scope-reconciliation code contains a confirmed shared defect: for plans reaching its scope-assignment branch, a keyword check matches "visible" without handling negation or exclusion. It therefore assigns viewport scope even for "outside the visible map," overwriting an otherwise appropriate filtered scope. This constraint is already recorded in the frozen task metadata and remains unchanged during evaluation. MD01 is consequently not a model-only failure. For these historical observations the shared rule is known, while raw pre-reconciliation plans are not retained; each model's original interpretation and additional contribution cannot be reconstructed. The matched GPT follow-up retains provider planner receipts: both models select filtered scope on all six MD01 attempts, after which the shared application changes it to viewport scope. This new evidence directly localizes that follow-up's six failures to the scope override; it does not recreate missing historical traces. The eighth campaign independently repeats this diagnosis across all nine MD01 attempts: every raw provider plan selects filtered scope, after which the unchanged postprocessor forces viewport and returns zero rather than the oracle's 266 reports.

MD09 accounts for the historical single-configuration API extension's other two failures. The application can compare original and estimated placements under the frozen task, but two responses select an unsupported operation. The third repetition passes the required evidence and caveat checks. This is a repeated-task inconsistency on a supported capability, not a transport error. All eight extension failures remain in the primary score and timing distribution.

In the matched GPT follow-up, GPT-5.4 chooses an unsupported operation on all three MD04 attempts, so those failures are not correct aggregate answers penalized only for missing annual rows. Sol returns the correct aggregate without annual rows twice and a correct annual breakdown once. On MD09, GPT-5.4 passes all three; Sol's retained raw planner chooses an unsupported operation all three times for a supported comparison. GPT-5.4 also refuses one supported MD07 query; Sol's sole MD07 non-success is instead the interrupted, unknown application outcome. The latter must not be attributed to model behavior or inferred from its completed provider output.

In the eighth campaign, Astra returns the correct MD04 aggregate in all three attempts but omits the annual rows (0/3); Sol supplies them once (1/3), and GPT-5.4 twice while refusing once (2/3). For MD07, Astra and Sol pass all three attempts, while GPT-5.4 selects unsupported or methodology plans and receives three HTTP 422 responses after the existing repair path. For MD09, GPT-5.4 passes all three, whereas Astra and Sol each refuse all three supported comparisons. GPT-5.4's single RV02 failure includes extra action fields equal to the current controls: strict action fidelity fails, while no actual unintended state change occurs. The inherited functional sensitivity credits that attempt, yielding 65/72 functional versus 64/72 strict successes. Astra and Sol remain 63/72 and 64/72 under that sensitivity. These distinctions prevent a shared total score from concealing different pipeline and contract failures.

In the lighter-model pair, Gemma also follows a selected-record path on MD01, whereas Qwen3.5 exhibits the viewport-scope failure. Gemma returns 422 on all three MD04 attempts; these are rejected queries, not correct aggregate answers penalized only for the annual-row ambiguity. Qwen3.5 instead passes MD04 in all repetitions. Its MD06 answers correctly explain missing ambient-weather coverage but use a disallowed breakdown operation; the secondary functional sensitivity credits these three responses. Its three MD17 failures preserve state but offer a generic refusal without the required City-versus-County clarification. Gemma fails MD15 by omitting the queryable-weather versus unavailable-filter distinction. These cases require different diagnoses despite similar total scores.

MD07 is a regression in the original Qwen pair: the baseline passes the selected-crash weather interpretation, while Qwen3.8 produces three 422 responses. MD12 shows the reverse pattern for clearing only the year restriction. Missing raw planner traces prevent a narrower diagnosis such as malformed JSON or a particular internal repair failure. The observable result is application validation failure on a supported task.

## 7.4 Narrative limitations remain part of correctness

The original Qwen-pair narrative review accepts 20 of 24 reviewed responses. Three failures are baseline control HIN11 responses missing the people-versus-crashes unit distinction in the HIN formula; one primary HIN12 response makes an unsupported claim about documentary or ranking consistency. The historical single-configuration API extension passes all twelve narrative

reviews under the inherited rubric; the matched follow-up passes all 24 reviews across its two models, and the eighth campaign passes all 36 reviews across its three models. The latter reviews were sealed before the Astra registration defect was diagnosed and remain unchanged. The ninth campaign accepts 47 of 48 reviewed narratives, with the single GPT-5.5 HIN14 claim-support failure explained in Section 6.3. The lighter-model campaign accepts 23 of 24 narrative assessments: all twelve Gemma responses and eleven Qwen responses. One Qwen HIN12 answer correctly notes conflicting influence-distance specifications and then makes an unsupported assertion that the dataset lacks such specifications. This failure concerns that claim, not an optional missing detail. These outcomes illustrate why source grounding includes units and uncertainty, not only matching numbers. They remain dependent on the stated AI-assisted review procedure and require human verification before strong domain conclusions are drawn. Passing a small known set of limitation questions does not establish general scientific reasoning ability.

# 8 Discussion

## 8.1 Deployment selection is a policy choice

A default model balances latency, observed task behavior, and the consequences of failure. The largest strict score does not automatically determine the best default for an interactive demonstration. For CERRA, the evidence supports retaining Gemma 4 12B as a practical local default while treating lighter Qwen3.5 9B as an optional candidate. The historical Qwen3.8 measurements remain scientific observations regardless of whether that slower configuration is exposed in the production selector. Gemma's historical 58/72 score, 3.51-second median, complete supported-filter performance, and zero observed UI-state violations informed that choice. The later paired campaign adds a fresh 60/72 observation with a 3.18-second median and 15/15 supported filter successes. Qwen3.5's 59/72 score, 2.95-second median, and smaller sampled residency make it a lighter candidate, but its repeated GPS-filter-reset omission argues against switching the default on latency alone. Qwen3.8 offers higher strict success in its own paired study with substantially longer latency.

This is an operational decision under incomplete evidence. There is no contemporaneous Qwen3.8–Gemma 4 paired experiment, no measured utility function, and no user study establishing a preferred waiting time. A focused UI follow-up and warm-session comparison would address gaps in the lighter-model study. The current default should be revisited using broader tasks and direct user feedback. We do not label it a statistically established optimum.

## 8.2 Design implications for analytical assistants

First, preserve state unless the requested action is both supported and unambiguous. A partial action can silently alter subsequent analysis even when the accompanying text acknowledges a limitation. Second, expose and test the boundary between a queryable attribute and an actionable filter; these capabilities need not coincide. Third, make source units and missingness explicit, particularly when users ask for risk, causality, or future outcomes beyond the evidence.

For evaluation, retain both contract-level and practical diagnostics. A strict score supplies a reproducible acceptance boundary, while mutation counts, HTTP outcomes, displayed-value checks, and reviewed narrative propositions explain what the score means. This recommendation follows from observed distinctions; the present study does not test whether a proposed UI redesign or new validator resolves them.

The earlier three-model API experiment supplies a separate operational observation: Astra's 2.49-second median is lower than the 3.82-second GPT-5.4 and 3.19-second Sol medians in that run, while its strict score is one attempt lower. Figure 2 adds marginal median-latency intervals; it does not estimate paired latency differences. Provider GPU allocation, energy use, and monetary cost are unmeasured, and the three pairwise score intervals are exploratory. This does not identify a universally preferable model or change the evidence for choosing among local GPU configurations. The post-run registration correction also demonstrates that validating a newly introduced identifier is a necessary part of preflight; preserving an older grader byte-for-byte did not guarantee a valid new-model evaluation.

## 8.3 Competitive scores do not imply a generational ranking

Release order alone does not predict performance on a fixed application contract. In the earlier three-model matched campaign, GPT-5.4 and Sol each pass 64/72 attempts and Astra passes 63/72; the pairwise intervals include or touch zero. The new four-model campaign records 63/72 for GPT-5.4, 68/72 for GPT-5.5, 63/72 for Astra, and 65/72 for Sol. GPT-5.5 has the highest point estimate in that run, but its +6.94-point difference from GPT-5.4 has interval $[-2.78, +19.44]$. The earlier GPT-5.4 observation is not a demonstration that older models are generally better, and the new GPT-5.5 observation is not evidence of universal superiority.

CERRA asks the model to select constrained operations and sometimes narrate supplied evidence; application code performs data queries, arithmetic, validation, and rendering. Good operation selection on these bounded workflows can yield a high score without testing capabilities that distinguish models elsewhere.

Only 24 distinct primary questions are repeated three times; repetition does not create 72 independent analytical tasks. Small score differences can depend on a few rubric-sensitive cases. In the latest run, all twelve MD01 plans select the correct scope before a shared downstream override causes failure. MD04's ambiguous annual-row requirement also affects separation among models. These constraints belong to the application and evaluation contract, not solely to foundation-model capability.

The new task-level results explain the observed ordering more precisely than version labels. GPT-5.5 succeeds on the supported MD07 queries that GPT-5.4 fails, and on MD09 comparisons refused by Astra and Sol; it also satisfies the MD04 annual-row requirement in every repetition. Its one HIN14 semantic failure comes from unsupported additional claims despite a correct causal refusal. These specific outcomes warrant domain review and a repaired, broader held-out suite. They do not identify a training-data, model-architecture, or provider-optimization cause.

All four configurations explicitly request medium reasoning and the same 8192-token combined budget. Thus the latest comparison does not contrast GPT-5.4 "ultra" against newer-model "medium" settings. Equal requested settings still do not measure equal provider compute or equivalent internal implementation of reasoning levels. Historical production GPT requests omitted reasoning effort and cannot be relabeled medium, ultra, or compute-matched after the fact. The chronology, requested settings, task-level behavior, and uncertainty must remain visible together.

### 8.4 Domain feedback and a prospective benchmark revision

Initial domain feedback identifies the current questions as closely coupled to an existing system, with insufficient evidence for generalization to broader interactive transportation analytics. We retain every historical task and score and narrow the paper to the CERRA case study. In particular, Baltimore-customized HIN products remain a historical case module; they are not proposed as a mandatory core for agencies that may lack equivalent data or methods. The three historical Chinese prompts are preserved for provenance, while the prospective v0.2 task inventory is English-only for its intended US transportation-professional audience.

The v0.2 review draft on CERRA proposes an English crash-analysis core and a separately gated infrastructure/asset-management extension. Its question inventory supports item-level review; CERRA sign-in is required. This is a preregistration draft, not an externally registered protocol. It contains 20 proposed crash-analysis questions, four separate controls, and 16 unscored expansion candidates. The initial core still targets Maryland application data and supplies no cross-jurisdiction validation. Candidate analytical questions should describe professional decisions and required evidence before being constrained to the current UI. Language-model suggestions can broaden the candidate pool, but do not certify relevance, feasibility, or correctness. Human reviewers must assess each item's analytical value, availability and licensing of data, required software and compute, verifiable answer or scoring rubric, and suitability for incorporation into the platform. Litter and invasive-species workflows belong in the asset-management branch and require their own data and operational validation; they are not scored by reusing crash-analysis outcomes. Other candidates span roadway exposure, pavement and bridge condition, travel-time reliability, transit supply, and accessibility. A source dataset or existing detector is not itself a validated fixture or ground-truth label set.

The proposed question contracts explicitly distinguish selection-wide, viewport, statewide, and selected-record scope, and require annual rows when annual breakdowns are intended. Prospective semantic scoring would accept alternative plans that return every requested quantity with correct scope, units, and evidence; it would not require one operation name when multiple plans are valid. Raw planner output, reconciliation changes, evidence queries, and final UI state would be retained separately to support failure attribution. These changes apply only to the next version: they do not relax historical strict criteria or reconstruct missing historical traces.

This is a design proposal, not an evaluated benchmark release. Candidate questions, proposed data requirements, and review worksheets are separated from the frozen historical manifests. Unsupported capabilities are recorded as implementation or resource gaps before testing, rather than being retrofitted into pass/fail criteria after seeing model answers. A new version requires reviewed English prompts, independent oracles, explicit permissible UI effects, a frozen execution and analysis plan, and a held-out assessment before claims of domain validity or cross-jurisdiction usefulness. Available application operations still require fixture-level preflight. In particular, C01 requires repair of the known negation-blind map-scope override and independent fixture validation before scored inclusion. Until then it remains an unscored proposal; any revised core and denominator must be frozen for all models before inference. Rewriting its question contract merely to avoid the defect does not establish a validated implementation. No v0.2 scores or generalization results are reported here.

## 9 Threats to Validity

**Task selection and coverage.** The suite is small, authored with application knowledge, and not sampled independently from user traffic or a transportation-

practice taxonomy. Its HIN questions depend on Baltimore-specific products; its three Chinese questions do not match the proposed English-only US-professional scope. Controls contain previously shown examples. Exact task reuse enables comparison but also creates a narrow fixed target. The study does not establish robustness to paraphrase distributions, multilingual use generally, adversarial inputs, or unseen analytical domains. Wide intervals reflect limited task coverage but do not correct selection bias. The prospective crash-core/asset-management redesign is untested, so it cannot repair the external validity of the results already collected.

**Configurations and execution conditions.** Weights, precision, parameter count, adapters, prompts, sampling, loading policy, and hosting conditions vary. Historical campaigns use different application deployments and a public network path. The original contemporaneous Qwen pair reduces some variation, but its models differ in size and offload on a shared GPU. The later Qwen3.5–Gemma pair differs in model packages and sampler defaults, and its residency telemetry is incomplete for Qwen. Cold-load timings do not estimate warm-session behavior. Neither later comparison supplies a matched Qwen3.5–Qwen3.8 contrast. The API extension uses a frozen local application, a regional remote endpoint, explicit medium reasoning effort, and a different combined reasoning/output budget. Those historical extension conditions do not supply a matched comparison with earlier API deployments or the Qwen pair. The later GPT follow-up matches application inputs and selected request settings within its own campaign, while leaving provider compute and alias drift unresolved. Its one interrupted application outcome is included as a primary non-success; missing timing and changed-weight sensitivity estimates are explicitly disclosed. The eighth campaign adds a third contemporaneous alias under the same requested settings, and the ninth adds GPT-5.5 alongside fresh observations of those models; their three and six pairwise intervals are unadjusted for multiplicity. The eighth campaign's post-run HIN identity-registration amendment changes the grading implementation while preserving task semantics; it is disclosed rather than represented as an unamended preregistration. Independent replay for that historical amendment finds no changed grades among the 1,260 earlier attempts. The ninth campaign carries the appropriate prior canonical identities and registers GPT-5.5 before freezing; new-versus-prior canonical replay leaves all 1,512 preceding attempts unchanged. This does not turn the historical eighth-campaign correction into a prospective change. Backend alias drift limits exact replay of hosted models. Timing includes loading and application work and is not a token-generation throughput measure.

**Measurement and adjudication.** The frozen rubric contains at least one wording/granularity ambiguity and distinguishes extra fields more strictly than visible harm. Narrative assessment lacks independent human raters. A corrected packet materially changed earlier judgments, and some enforcement code was added after the written protocol freeze. These changes are documented, but the work should not be described as an externally preregistered trial. Structural graders do not exhaustively validate every surplus prose assertion or source label.

**Attribution and statistical scope.** Shared preprocessing and postprocessing can dominate outcomes. Final artifacts do not always include raw model intermediates. The study therefore attributes effects to configurations and avoids isolated causal claims. Bootstrap intervals are unadjusted and dependent on this convenience suite; the number of repetitions is too small to characterize rare failures or long-run consistency precisely. No inference cost, energy consumption, human productivity benefit, or crash-risk reduction was measured.

# 10 Reproducibility and Research Practice

The retained research archive contains frozen manifests, exact contexts and histories, source-derived oracles, code and data hashes, package identifiers, request outcomes, browser evidence, review decisions, and additive audit notes. Appendix B identifies the campaign manifests. Aggregated figures in this draft are reproducible from the included CSV/JSON and plotting scripts; they introduce no new inference runs. The added median-latency intervals are computed from retained timing records and bundled as sanitized aggregate endpoints with source hashes and the bootstrap specification; the figure-rendering package does not redistribute private per-attempt timing records.

The public CERRA benchmark artifact (v0.1.0) provides all 28 original question strings, fixed contexts, scoring rules, frozen grader code, relevant application snapshots, and sanitized per-attempt outcomes supporting offline aggregate reconstruction for the first five campaigns (924 attempts). That frozen release does not contain the sixth Qwen3.5–Gemma, seventh matched GPT, eighth three-model GPT, or ninth four-model GPT campaign. The updated manuscript package provides their sanitized aggregates, paired summaries, runtime coverage or interruption sensitivities, and source hashes. The eighth campaign additionally records the identity-registration amendment, original and amended grader hashes, and all three exploratory contrasts. The ninth records prospective model registration, all six exploratory contrasts, 48 sealed nar-

rative judgments, and prespecified median-latency intervals. Its new plot records preserve the original eighteen observations and intervals unchanged. The separately linked v0.2 review draft describes proposed tasks without results; its version hashes are retained in `benchmark-design-provenance.json`. Full per-attempt public coverage remains limited to v0.1.0 until a separately versioned extension is released. A questions-first review packet includes answer keys, known ambiguities, and a blank feedback worksheet; independent item-level domain validation remains pending. One selected-record task (MD07) uses a documented public placeholder. Exact end-to-end reruns require the retained source data, record binding, application environment, and browser harness; raw answers, individual source records, credentials, and private operational traces are not redistributed. The website remains access-controlled. The first-page badge links to the project-specific ARA, which records bounded claims and their evidence. Code and review materials are MIT-licensed; the manuscript retains author rights.

This draft was assembled with AI assistance from experiment artifacts. Authorship, affiliations, funding, acknowledgments, disclosure wording, and the target venue require author confirmation. Domain feedback has informed the scope revision and prospective task design. Independent item-level domain validation, adjudication of narrative answers, and a fresh held-out evaluation remain unfinished; the manuscript does not represent them as completed.

## 11 Conclusion

Evaluating a language-model assistant inside an analytical dashboard requires checking the complete workflow: intended data scope, independent evidence, required qualifications, and actual state changes. Local CERRA comparisons expose both a quality–latency trade-off and a lighter configuration without established equivalence. Earlier matched API campaigns retain their own settings, scores, interruption sensitivity, and identity-registration audit history. The latest four-model campaign compares GPT-5.4, GPT-5.5, Astra, and Sol under the same exposed medium reasoning and combined output budget, recording 63/72, 68/72, 63/72, and 65/72 strict successes. GPT-5.5 has the highest observed score, while Astra has the shortest observed median request latency. All six exploratory pairwise score intervals include or touch zero, and separate marginal latency intervals do not establish a paired speed difference or a universally preferable model. Task-specific refusals, an ambiguous annual-row contract, a shared scope-reconciliation defect, and one unsupported narrative addition explain why aggregate ordering cannot substitute for failure attribution. These are bounded CERRA observations, not evidence of generic transportation competence or cross-jurisdiction transfer. The proposed English crash-analysis core and asset-management extension remain untested. Reporting requested reasoning settings, task outcomes, response-time uncertainty, and audit integrity together preserves the case study's value without overstating either a model generation or an unevaluated redesign.

## References

[1] P. Liang et al. Holistic Evaluation of Language Models. *Transactions on Machine Learning Research*, 2023. https://openreview.net/forum?id=iO4LZibEqW.

[2] X. Liu et al. AgentBench: Evaluating LLMs as Agents. In *International Conference on Learning Representations*, 2024. https://arxiv.org/abs/2308.03688.

[3] S. Zhou et al. WebArena: A Realistic Web Environment for Building Autonomous Agents. In *International Conference on Learning Representations*, 2024. https://arxiv.org/abs/2307.13854.

[4] S. Yao, N. Shinn, P. Razavi, and K. Narasimhan. $\tau$-bench: A Benchmark for Tool-Agent-User Interaction in Real-World Domains. In *International Conference on Learning Representations*, 2025. https://arxiv.org/abs/2406.12045.

[5] S. Kapoor, B. Stroebl, Z. S. Siegel, N. Nadgir, and A. Narayanan. AI Agents That Matter. *Transactions on Machine Learning Research*, 2025. https://openreview.net/forum?id=Zy4uFzMviZ.

[6] Maryland Department of State Police. Interactive Data Dashboards. Accessed September 29, 2026. https://mdsp.maryland.gov/safety-prevention/interactive-data-dashboards.

# A Task Inventory

Descriptions below are compact paraphrases, not replacements for the frozen prompts and initial contexts. A dagger marks one of the three Chinese prompts, translated here for readability. "Browser" denotes an observed UI task; other tasks use the application API. Controls remain outside all primary score and latency denominators.

| ID | Family | Mode | Task and principal acceptance boundary |
|---|---|---|---|
| MD01 | Data query | API | Count the full current crash selection, including reports outside the visible map. |
| MD02 | Data query | API | Summarize statewide ambient-weather completeness across all years. |
| MD03 | Data query | API | Produce statewide county-by-county totals for 2021, without severity restriction. |
| MD04 | Data query | API | Tabulate Baltimore County totals for all years and severities; frozen oracle requires annual rows (wording ambiguity noted). |
| MD05 | Data query | API | Count 2022 statewide crash reports listing ambient rain. |
| MD06 | Data query | API | Resolve a follow-up "What about 2025?" using the frozen canonical history. |
| MD07 | Data query | API | Explain whether an unmarked weather-contributor field on a selected crash establishes clear weather. |
| MD08 | Data query | API | Describe GPS screening for visible crashes without overstating positional certainty. |
| MD09 | Data query | API | Compare original and estimated locations in the viewport and count reports entering or leaving. |
| MD10† | UI action | Browser | Filter 2021 Baltimore County serious-injury crashes. |
| MD11 | UI action | Browser | Filter 2022 Anne Arundel fatal or serious-injury crashes. |
| MD12 | UI action | Browser | Remove only the year restriction; preserve other selections. |
| MD15 | UI boundary | Browser | Handle unsupported rain/fatal filtering; explain the query/filter distinction and preserve state. |
| MD17 | UI boundary | Browser | Clarify ambiguous "Baltimore" before acting on a 2020 map request. |
| RV01 | UI action | Browser | Change only county and severity; preserve year, GPS, estimates, and map positions. |
| RV02 | UI action | Browser | Clear only GPS cross-check while preserving estimate filters and comparison positions. |
| HIN02 | HIN source | API | Return the three half-mile-minimum corridors with the largest source fatality counts. |
| HIN03 | HIN source | API | Return the first three Tier 2 intersections in source-rank order. |
| HIN05 | HIN source | API | Rank four tracts by households without vehicles, excluding unavailable values. |
| HIN06† | HIN source | API | Count all Major Concern severe-injury source records, ignoring current map filters. |
| HIN10† | HIN source | API | Resolve a follow-up switching to the two-mile-minimum corridor group, retaining top-three scope. |
| HIN12 | HIN limits | API | Qualify conflicting intersection influence distances and differences between public and application rankings. |
| HIN13 | HIN limits | API | Decline an unsupported forecast of next-month fatal-crash probability per vehicle-mile. |
| HIN14 | HIN limits | API | Decline an unsupported causal percentage reduction from a hypothetical North Avenue road diet. |
| MD16 | Control | Browser | Deterministic refusal: rain cannot be established as causing fatal crashes per mile from these data. |
| MD18 | Control | Browser | Previously shown example: 2023 Baltimore City fatal-crash filtering. |
| HIN01 | Control | API | Previously shown example: retrieve the five highest-ranked one-mile corridors. |
| HIN11 | Control | API | Previously shown example: explain HIN scoring, including people-versus-crashes units. |

# B Provenance and Audit Inventory

The aggregate package is derived from the final audited summaries of the following retained campaigns. Campaign and model hashes are listed in `provenance.json`; source-export hashes are listed in `data-provenance.json`; abbreviated hashes below are human-readable identifiers, not substitutes for full integrity checking. Dates are local Maryland campaign labels, not unique experiment identifiers. The original local Qwen pair, API extension, and lighter-model pair have distinct campaign folders and manifest hashes despite sharing a local date.

| Campaign | Archive directory | Trials | Manifest SHA-256 prefix | Context |
|---|---|---|---|---|
| Sep. 18 | `transportation-model-benchmark-2026-09-18` | 252 | `a73f5b9e835ec776` | Production v36 |
| Sep. 23 | `transportation-gpt-benchmark-2026-09-23` | 252 | `9239d992da1869f9` | Production v48 |
| Sep. 24 | `local-model-benchmark-2026-09-24` | 168 | `a2fb59169c45f215` | Production v51/v52 |
| Sep. 29 | `qwen-newer-benchmark-2026-09-29` | 168 | `a32cb875e9e9b2ac` | Compiled local app |
| Sep. 29 | `gpt61-sol-benchmark-2026-09-29` | 84 | `b2777e4a29be6c55` | Frozen local app / API |
| Sep. 29 | `qwen35-benchmark-2026-09-29` | 168 | `4716c4dfc609ebf2` | Lighter-model local pair |
| Sep. 30 | `gpt-matched-benchmark-2026-09-30` | 168 | `45249e285c585719` | Matched local app / API |
| Sep. 30 | `gpt6-threeway-benchmark-2026-09-30` | 252 | `de9b252a331c78eb` | Three-model matched API |
| Oct. 2 | `gpt55-fourway-benchmark-2026-10-02` | 336 | `79359ef7cf632123` | Four-model matched API |

All three September 29 campaigns use application base `6ed65fc06f525e8bbe8cb800ce80487642f6120a`, with separately retained routing/configuration changes. The paired local run spans 14:17:05–14:50:59 UTC; the API extension spans 23:50:47–23:58:05 UTC on September 29, 2026. Qwen3 14B package digest begins `bdbd181c33f2ed1b`; Qwen3.8 27B Q4_K_M begins `25b843619e944cd0`. The lighter-model pair spans 01:45:20–02:01:53 UTC on September 30 (September 29 EDT). Its Qwen3.5 9B package digest begins `6488c96fa5faab64`; the Gemma 4 12B digest begins `4eb23ef187e2c546`. Display labels are rounded: the package metadata reports 9.7B and 11.9B parameters, respectively. The API extension records the provider-reported alias and frozen runtime settings rather than a weight digest. The matched API campaign uses the same application base with its own transport settings and manifest; it spans 01:37:05–01:58:57 UTC on October 1 (September 30 EDT). Of its 168 settled attempts, one application outcome is interrupted and unknown; the retained amendment distinguishes it from the 167 HTTP-200 responses and excludes its unmeasured completion time. The eighth campaign spans 02:39:23–03:05:32 UTC on October 1 (September 30 EDT), retaining all 252 outcomes and 36 reviewed narratives. Its 282 scored provider calls are separate from six non-scored preflight calls. Its original HIN grader hash begins `bc9827d46a6bce48`; the registered derivative begins `aaec3d8ac602a650`. The original 33 frozen dependencies remain intact; the derivative and amendment are separately retained. The ninth campaign runs from 19:20:17 to 19:55:53 UTC on October 2 (October 2 EDT), retaining all 336 application outcomes and 48 reviewed narratives. Its 376 scored provider calls are separate from eight non-scored preflight calls. GPT-5.5 and Astra are prospectively registered in its canonical grader; no post-run identity amendment occurs. Final verification accounts for all scheduled attempts, browser records, and narrative decisions. Package digests identify tested local artifacts; family names alone do not.

**Audit distinctions to preserve when revising.** Keep the contemporaneous 49/72 Qwen3 14B baseline separate from its historical 51/72 observation, and the later 60/72 Gemma result separate from its historical 58/72 result. Do not treat the later Qwen3.5 comparison as a matched test against Qwen3.8, or sampled residency as peak memory. The public v0.1.0 release retains its original five-campaign scope. Keep all three matched GPT campaigns separate from historical rows and from one another. The ninth campaign prospectively registers GPT-5.5 and retains all six unadjusted pairwise contrasts; its higher GPT-5.5 point estimate is not a general ranking. Retain the eighth campaign's explicit Astra registration amendment: its initial 39/72 and 6/12 registry-artifact totals become 63/72 and 12/12, without treating the initial totals as model performance. Report all three pairwise contrasts and their unadjusted status. For the seventh campaign, retain the interruption in the primary denominator, and do not substitute a provider or administrative timestamp for missing application latency. Do not merge control results with primary results, apply the primary uncertainty interval to secondary scores, call 422 responses network outages, or infer raw planner behavior from final postprocessed records. Narrative review is AI-assisted, and the September 24 correction is part of the record. Data correctness concerns the retained snapshot, not external epidemiological validity.

**Author review before submission.** Confirm author order and affiliations; verify all narrative judgments with independent domain reviewers; choose a venue and adapt its template and disclosure rules; decide what sanitized evaluation artifacts can be released; and add a held-out evaluation if claiming generalization beyond this

implementation-aware suite. These items describe unfinished research or administrative work, not completed results.