EAB: AN EMBODIED AGENT BENCHMARK WITH ANCHORED CRITERIA AND PAIRED INTERVENTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Embodied agents fail mid-task: a blocked path, an injected fault, a user interrupt. Outcome-level scoring over dialogue transcripts neither sees these failures nor attributes them. We present EAB, a benchmark for embodied agents at the text-and-tool control layer, whose judgment is anchored to execution. Its 400 cases, set in a tour-guide robot domain with 29 skills on 4 MCP servers, 10 real-domain error codes, and 6 mid-task event classes, form 200 strictly matched pairs, each built around one ablated variable, so the within-pair difference estimates that variable's effect within the benchmark. Verdicts are three-state over 25 criteria aggregated along two axes, four report families and 14 capability units. NA, whether designed away by the case's criteria scope or never triggered by the episode, is reported as its own verdict and excluded from every denominator. Grading reads execution facts within per-case anchor windows, and the pipeline is validated by golden samples, mutation-tested gates, and a replay ceiling. Evaluating frontier LLM agents under three drive modes (simulator, solo, prescribed), we find that interaction with a user, rather than a skill-level plan, accounts for most of the difficulty: for the three agents run in every mode, only 4.2–8.2% of cases pass all three repeats under interaction, against 38.8–54.0% without a user (0.5–9.8% across all eight agents under interaction). The benchmark then localizes where they break. Terminal success under interaction is comparatively high (43.0% Avg@3 for the reference agent), yet the same episodes satisfy every mounted scored criterion only 8.5% of the time, so the failures are mid-task ones that outcome-scored benchmarks do not see.

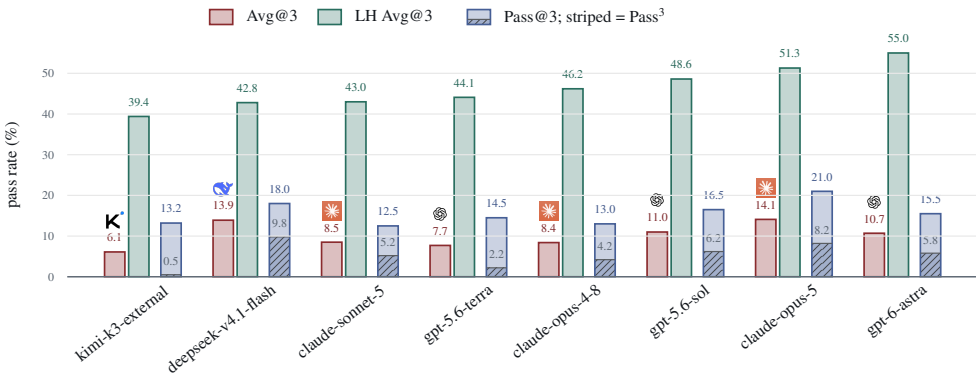


Figure 1: Overall performance on EAB in simulator mode across eight agents (k=3, 400 cases; identical harness, user simulator, and judge): case-level Avg@3, per-case Pass@3 with Pass^3 as the striped lower segment, and the long-horizon channel Avg@3 (`goal_passed` ∧ `final_state_match` in the same run), sorted by the long-horizon channel. Compliance under interaction, not task success, is the bottleneck.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

LLM-based agents are increasingly deployed as embodied controllers: they plan multi-step tasks, call tools, and act on a physical or simulated world. When such an agent fails mid-task—a blocked path, a fault injected by the environment, a user interrupting with a changed instruction—the failure is an *execution* failure: it lives in the sequence of tool calls, results, replies, and state transitions, not in the final utterance. Yet the dominant evaluation paradigm grades final answers over dialogue transcripts.

The gap. Tool–agent–user benchmarks such as τ-bench, τ^2-bench, and VitaBench (Yao et al., 2025; Barres et al., 2026; He et al., 2026) grade task outcomes: database assertions and reward functions over final state, occasionally with policy-compliance checks. Rubric-based evaluators decompose trajectories into checklists but score them with LLM judges whose verdicts are neither anchored to specific execution events nor validated against known ground truth. Embodied benchmarks—from vision-and-language navigation and household simulation (Shridhar et al., 2020; Padmakumar et al., 2022; Puig et al., 2018) to recent human-in-the-loop and LLM-planner suites (Puig et al., 2024; Li et al., 2022; Chang et al., 2025; Cheng et al., 2025)—grade action sequences or predicate satisfaction in perceptual simulators, but rarely grade tool behavior or attribute failures to specific capabilities. Across all three lines, two structural problems recur. First, *unanchored judgment*: a criterion such as "the agent should acknowledge the interrupt" is checked against the whole transcript, so when it fails, one cannot tell whether the agent ignored the interrupt, or acknowledged it outside the window the designer intended. Second, *confounded variables*: a low score on tasks with injected faults mixes at least three causes—the agent cannot use the tools, the agent does not handle the fault, and the task is simply harder—and no standard benchmark can separate them, because task difficulty and the variable of interest are not varied independently.

Each desideratum has a precedent in isolation—controlled single-variable comparison in contrast sets (Ribeiro et al., 2020; Gardner et al., 2020), evidence-bound grading as a response to judge noise (Zheng et al., 2023), and validated pipelines in benchmark audits (Reuel et al., 2024)—but no prior benchmark combines anchored judgment, paired interventions, and a validated harness; in Table 1, no prior benchmark fully satisfies any of the three.

EAB. We present EAB (Embodied Agent Benchmark), a benchmark for text-and-tool embodied agents in a tour-guide robot domain, built on three principles. *Anchored criteria*: 25 criteria (23 in four report families; the other two on non-scored channels, Sec. 3.5) issue three-state verdicts (pass / fail / NA) by reading execution facts—tool calls, tool results, replies, and state timelines—within per-case anchor windows, binding each verdict to the events it is about. *Paired controlled interventions*: all 400 cases form 200 strictly matched pairs, 183 of which ablate one environment variable, so the within-pair difference Δ_p estimates that variable's effect within the benchmark without cross-task confounds; the other 17 change only a scoring rule while the agent's inputs stay byte-identical, and serve as a test–retest null control for the estimator. *A validated harness*: run–grade separation, golden samples, mutation-tested gates, a deterministic replay ceiling, and a fingerprinted dataset manifest make the pipeline itself an audited artifact.

EAB's sandbox enforces a single owner for world state, exposes 29 skills across 4 MCP servers (Anthropic, 2024) with 10 real-domain error codes, and injects 6 classes of mid-task events (user queries, state queries, state probes, user interrupts, async notifications, task conflicts). Three drive modes (simulator, solo, prescribed) place the agent under user-in-the-loop, independent, and plan-given conditions; a deterministic replay harness (ceiling) validates the benchmark itself. Scores aggregate along two axes—four report families and 14 capability units (11 ablatable)—so a failure can be attributed to a family, a unit, or a specific ablated variable. Fig. 1 previews the operative finding of Sec. 4: under the user-in-the-loop, behavioral compliance—not terminal task success—is where frontline agents break; Fig. 2 gives an overview.

Contributions.

1. **EAB**: 400 cases (200 strictly matched pairs) in a single-owner-state sandbox with 29 skills across 4 MCP servers, 10 error codes, and 6 mid-task event classes (Sec. 3).
2. **Anchored criteria judgment**: 25 three-state (pass/fail/NA) criteria (23 in four report families), graded on execution facts within per-case anchor windows; NA is first-class (designed vs. untriggered) and reported separately from pass/fail (Sec. 3.5).

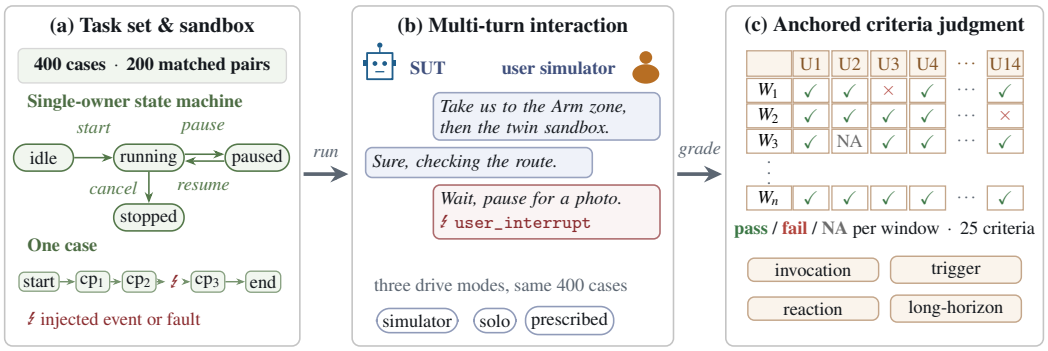


Figure 2: EAB overview. (a) Task set and sandbox: 400 cases forming 200 strictly matched pairs over a single-owner state machine; each case is a goal, a checkpoint chain, injected events, and ground truth. (b) Multi-turn interaction: the SUT converses with a goal-driven user simulator and calls skills under three drive modes. (c) Anchored criteria judgment: per-case windows over execution facts yield three-state verdicts on 25 criteria, along family and unit axes.

3. **Paired controlled interventions**: 183 controlled pairs are each built around one ablated environment variable, so the within-pair difference Δ_p estimates that variable's effect within the benchmark without cross-task confounds; 17 null-control pairs, identical on the agent side, calibrate the estimator's sampling noise, with Δ_p indistinguishable from zero for all eight agents (Sec. 3.4).
4. **A validated, reproducible harness**: run–grade separation, golden samples, mutation-tested gates, a replay ceiling, and a fingerprinted manifest ("same fingerprint, comparable scores") (App. E).

2 Related Work

Tool-using LLM agent benchmarks. τ-bench (Yao et al., 2025) introduces the tool–agent–user interaction setting, grading database state against policy compliance after each episode; τ^2-bench (Barres et al., 2026) extends it to dual control; VitaBench (He et al., 2026) scales to a life-services domain with an LLM-judged rubric sliding window. ToolSandbox (Lu et al., 2025) adds stateful conversational tool use with milestone and minefield assertions; AppWorld (Trivedi et al., 2024) provides executable apps with unit tests and contrast-set task variants; MCP-Universe (Luo et al., 2025) evaluates agents against real-world MCP servers. A broader family—BFCL (Patil et al., 2025), SWE-bench (Jimenez et al., 2024), AgentBench (Liu et al., 2024), WebArena (Zhou et al., 2024), OSWorld (Xie et al., 2024), AndroidWorld (Rawles et al., 2025), WorkArena (Drouin et al., 2024), GAIA (Mialon et al., 2024), TheAgentCompany (Xu et al., 2025)—shares the same paradigm. All grade *outcomes*: final state, final answer, or LLM-judged checklist coverage; none binds individual criteria to declared *event* windows with three-state verdicts, and none builds controlled ablation into the data itself (Table 1).

Rubric- and judge-based evaluation. LLM-as-a-judge is convenient but noisy (Zheng et al., 2023), and rubric checklists over transcripts inherit that noise. EAB instead states criteria as procedures over execution facts inside declared windows: 17 deterministic matchers, plus 8 constrained yes/no LLM sub-questions over in-window evidence (App. C)—rubric evaluation as audited code.

Embodied and interactive benchmarks. Classic embodied benchmarks evaluate instruction following in visual simulators—ALFRED (Shridhar et al., 2020) and TEACh (Padmakumar et al., 2022) over AI2-THOR, VirtualHome (Puig et al., 2018) over program-driven households—grading action sequences and final task state. The LLM era has scaled both simulators and tasks: Habitat 3.0 (Puig et al., 2024) puts real humans in the loop; BEHAVIOR-1K (Li et al., 2022) defines 1,000 everyday activities; PARTNR (Chang et al., 2025) generates 100K human–robot collaboration tasks; EmbodiedEval (Cheng et al., 2025) and LoTa-Bench (Choi et al., 2024) evaluate multimodal and language-oriented planners. Grading here is predicate- or state-based over whole episodes, with perception the dominant difficulty. EAB occupies the complementary regime of text-and-tool con-

Table 1: EAB in the landscape of agent benchmarks: embodied benchmarks over 3D visual simulators (top block) and tool-using benchmarks over digital worlds (bottom block). The top block is listed for positioning only: those benchmarks target perception and motor control, which EAB leaves out of scope, so its columns are not a like-for-like comparison. ✓ = yes, ◐ = partial, × = no. Row labels carry citations; all are discussed in Sec. 2.

Benchmark	User sim.	Stateful	Events	Exec-fact	Anchored	Paired	Validated
Embodied benchmarks (3D visual simulators)							
ALFRED [Shridhar et al., 2020]	◐	✓	×	✓	×	×	×
TEACh [Padmakumar et al., 2022]	◐	✓	×	✓	×	×	×
BEHAVIOR-1K [Li et al., 2022]	×	✓	×	✓	×	×	×
Habitat 3.0 [Puig et al., 2024]	◐	✓	×	✓	×	×	×
PARTNR [Chang et al., 2025]	◐	✓	×	✓	×	×	×
LoTa-Bench [Choi et al., 2024]	×	✓	×	✓	×	×	×
Tool-using agent benchmarks (digital worlds)							
τ-bench [Yao et al., 2025]	✓	✓	×	✓	×	×	×
ToolSandbox [Lu et al., 2025]	✓	✓	◐	✓	◐	×	×
AppWorld [Trivedi et al., 2024]	×	✓	×	✓	×	◐	◐
BFCL [Patil et al., 2025]	◐	◐	×	✓	×	×	×
OSWorld [Xie et al., 2024]	×	✓	×	✓	×	×	×
τ^2-bench [Barres et al., 2026]	✓	✓	◐	✓	×	×	×
VitaBench [He et al., 2026]	✓	✓	◐	×	◐	×	◐
MCP-Universe [Luo et al., 2025]	×	✓	×	✓	×	×	×
EAB (ours)	✓	✓	✓	✓	✓	✓	✓

Columns encode the desiderata of Sec. 1: a goal-driven user simulator; a persistent stateful world; mid-task events (faults, interrupts, async notifications) as first-class stimuli; execution-fact grading from traces rather than LLM-judged text; criteria anchored to declared event windows; paired interventions as controlled single-variable case pairs; a validated pipeline (golden samples, mutation-tested gates). *Partial marks*: in the embodied block, ◐ on user-sim. denotes human or scripted/LLM-driven *partners* rather than a dialogue simulator; anchored ◐ denotes the closest prior mechanisms (ToolSandbox milestones; VitaBench's sliding-window rubric), neither binding per-criterion verdicts to declared *event* windows.

trol: the world is a state machine, not a renderer, so the hard part is decisions over tools and mid-task events and grading reads traces rather than pixels.

Controlled interventions. Prior benchmarks ablate at analysis time—authors re-run tasks with a knob changed. EAB freezes the intervention into the dataset: each pair is identical except one declared variable, so any user measures the same contrast from the released data—comparable across papers.

3 The EAB benchmark

Table 3 (App. A) collects the notation used throughout.

3.1 Design principles

Three principles respond to the gaps of Sec. 1. **C1, anchored judgment**: every criterion reads execution facts inside a declared window, so a verdict traces to the events it is about, unlike outcome assertions (Yao et al., 2025; Trivedi et al., 2024) or rubric windows (He et al., 2026). **C2, paired interventions**: 183 controlled pairs each ablate one environment variable and are released as data, so effects are estimable without cross-task confounds. **C3, a validated harness**: the grader is code with tests—17 deterministic matchers plus 8 constrained yes/no judge sub-questions—validated by construction (golden traces, mutation-tested gates, deterministic replay) rather than by agreement with a free-form judge (He et al., 2026; Zheng et al., 2023). Replaying golden actions scores 0.991 with zero evidence gaps, but five event-reaction criteria fall short of 1.00 (lowest: `confirm_before_irreversible`, 0.52); dropping all five leaves every significant paired effect significant with the same sign (5/5) and moves simulator Avg@3 from 8.5% to 10.1% (App. E).

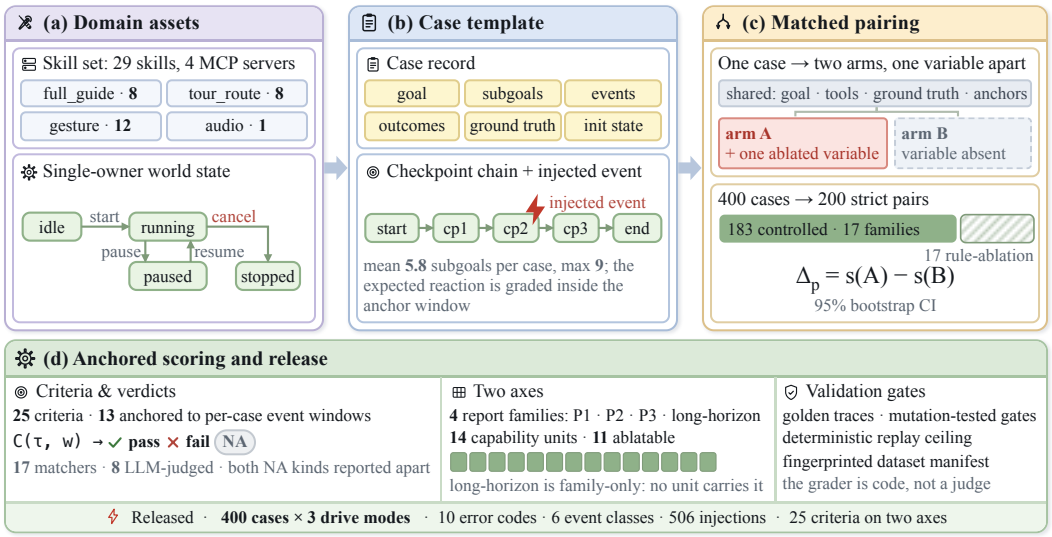


Figure 3: EAB construction pipeline. (a) Domain assets: skills on mock MCP servers, a single-owner state machine, and an injection library of error codes and mid-task events. (b) Each case: a goal with checkpoint subgoals, initial state, triggered events, injected tool outcomes, and ground truth. (c) 400 cases form 200 strictly matched pairs one variable apart, so the within-pair difference estimates that variable's effect. (d) Anchored three-state scoring on two axes, under validation gates.

3.2 Task definition and environment

An EAB *case* is a tuple $c = \langle g, s_0, \mathcal{T}, \mathcal{E}, \mathcal{O}, \mathcal{G} \rangle$ (Fig. 3b): a goal g decomposed into checkpoint subgoals (mean 5.8, max 9); an initial state s_0; the skill set $\mathcal{T}$; triggered events $\mathcal{E}$; injected tool outcomes $\mathcal{O}$; and ground truth $\mathcal{G}$—expected tool chain, expected final state, and the *criteria scope* (which criteria mount, at which anchors, with what expected reactions). The domain is a tour-guide robot sandbox with a single owner for world state: **29 skills across 4 MCP servers** (Anthropic, 2024), **10 real-domain error codes** as tool outcomes, and **6 classes of mid-task events** (user queries, state queries, state probes, user interrupts, async notifications, task conflicts) for 506 injections. App. B.1–B.3 detail the dataset, its families, and its injections.

3.3 Drive modes and harness

EAB provides **three drive modes** (simulator, solo, prescribed), plus **ceiling**, a deterministic replay harness used only to validate the benchmark and excluded from rankings. *Simulator* mode puts the agent with a goal-driven LLM user simulator under a 30-turn budget; *solo* mode has it work autonomously from a written brief under a shorter 8-turn budget (no interlocutor must be negotiated with); *prescribed* mode gives the correct plan—skill sequence and rationale, *without* argument values—at the first turn, on the same 30-turn budget as simulator. Unlike the oracle-plan ablation of τ^2-bench (Barres et al., 2026), which hands over the full ground-truth action sequence, prescribed withholds argument values, so parameters must still be elicited and injected events stay out-of-plan. The modes bound the evaluation axis (full interaction / no user-management variance / no planning variance), and every case ships all three payloads: $400 \times 3 = 1{,}200$ episodes per repetition, or 3,600 per system under test at full coverage (Sec. 4.1).

3.4 Paired controlled interventions

All 400 cases form **200 strictly matched pairs**: **183 controlled pairs** over **17 environment-variable families**, each built around one ablated variable so the within-pair difference estimates that variable's effect within the benchmark. The variable necessarily mirrors into what the agent sees (task text 39/200 pairs, reference dialogue 125/200, per-mode payloads 169/200), so Δ_p measures the variable as realized in the released cases, not an abstract semantic factor (App. B.4). The remaining 17 **rule-ablation pairs** are a deliberate test–retest null control: both arms give the agent and the user simulator byte-identical inputs (the task text itself asks for the stops in order), and

only the grader's ordering rule flips (strict → free), so their true effect on shared criteria is zero by construction. They never enter effect estimates; they measure how far Δ_p drifts from zero under sampling noise alone (18 variables in total). A pair toggles a confirmation requirement before an irreversible call, a mid-tour navigation fault, or an out-of-scope request, holding goal, tools, and anchor events fixed (Fig. 7); effects are reported per family with bootstrap intervals whose resampling unit is the pair (pass counts pooled over the 3 repeats within each arm). Intervals are not corrected for multiple comparisons across families and agents, so we read the paired effects as exploratory.

3.5 Anchored criteria and dual-axis scoring

EAB grades with **25 criteria, 23 of them in four report families** (the remaining two sit on non-scored channels below): invocation correctness (P1: the right tools, right arguments), trigger correctness (P2: the right tools at the right time), reaction to anomalies (P3: behavior on faults, interrupts, and conflicts), and long-horizon progress (checkpoint chains over multi-phase tasks). Formally, an anchored criterion is a procedure $C(\tau, w) \to \{\text{pass}, \text{fail}, \text{NA}\}$ mapping an execution-facts trace τ and an anchor window $w = [t, t{+}N)$ to a three-state verdict, reading only facts inside w: an interrupt-obedience criterion reads the turns immediately after the interrupt event, not the whole transcript. 13 of the 25 criteria are anchored (Fig. 8, App. B.5); the rest scan fixed ranges (e.g., every tool call, every turn, the final state). VitaBench's sliding-window rubric (He et al., 2026) is the closest evaluator, but its windows are scanned by a judge over free text, whereas EAB declares them per case and reads them by code: 17 deterministic matchers plus 8 constrained yes/no sub-questions for an LLM judge (App. C, D.3).

NA is first-class: *designed-NA* (the case's criteria scope does not mount the criterion) and *untriggered-NA* (mounted, but the episode never presents the stimulus it scores—a fault-reporting obligation with no fault, an interrupt criterion with no user). Both are reported as separate verdict classes, never as pass or fail. Family scores are per-case pass rates: a case passes a family iff every mounted verdict of that family passes, over the cases for which the family mounts at least one scored verdict; NA is excluded from the denominator throughout. Below, *pair* always means a matched case pair; a mounted (case, criterion) slot is a *verdict*.

Scores aggregate along two axes: a **family axis** (P1–P3, long-horizon) for reporting and a **capability axis** of **14 units** (11 ablatable; U1/U2 always-on, U14 mounted per case by stated obligation) for attribution. Both read the same criteria, with one deliberate exception: the long-horizon family is the two case-level terminal criteria, which belong to no unit, so unit-scored verdicts cover P1–P3 in full and long-horizon success is reported on the family axis only. Two further channels carry a tracked criterion each (turn budget, and a diagnostic readout of reply grounding), and a third (safety) is a red-line channel scored zero-tolerance outside the main case set; none of the three enters case-level pass.

4 Experiments

4.1 Setup

We evaluate eight frontier LLM agents (App. D.1), each case run as an isolated process under a per-episode turn budget.[1] Every SUT runs unmodified inside the same production embodied-agent gateway that ships with the harness (its README records the exact build); only the served model changes across the matrix. All runs use the fingerprinted EAB v1 dataset: 400 cases × 3 modes × 3 repeats = 3,600 episodes for the three Claude agents with full mode coverage (*claude-sonnet-5*, *claude-opus-4-8*, *claude-opus-5*); the other five (*gpt-5.6-sol*, *gpt-6-astra*, *gpt-5.6-terra*, *deepseek-v4.1-flash*, *kimi-k3-external*) run in simulator mode only (1,200 episodes each). The analyses below report the reference agent (*claude-sonnet-5*) in full; Sec. 4.5 adds the cross-agent comparison. Two endpoint constraints make the protocol differ by family (App. D.1): GPT endpoints reject `temperature=0` and run at the endpoint default, and the Claude gateway does not enforce `enable_thinking=false`, so the Claude agents run with thinking in effect. Cross-family statements therefore concern patterns (mode gaps, break points, paired effects), not absolute scores, and Fig. 1 is not a leaderboard.

[1] Full per-role configuration and prompts are in App. D; the harness, dataset, and fingerprinted manifest are in the supplementary material.

We report three complementary case-level metrics over k=3 repeats, following Yao et al. (2025): **Avg@3** (macro mean of per-case pass rates), **Pass@3** (at least one pass in k tries), and **Pass**^{3} (all k passes, a reliability lower bound), both computed per-case over the 400 cases. An episode passes iff every mounted scored criterion passes; NA never counts as pass or fail, and the two terminal criteria (`goal_passed`, `final_state_match`) form a separate long-horizon channel rather than gating case pass, as do the non-scored channels. Criterion-level scores are per-case pass rates over mounted criteria with NA excluded (Sec. 3.5). The grading implementation and its validation are detailed in App. E.

4.2 Main results: how frontier agents fare on EAB

Table 2 collects family-level, case-level, and long-horizon metrics per drive mode; the capability-unit axis is in App. C (Table 9); both opus agents are in Table 13. Four findings hold for the reference agent.

F1: Interaction with a user, rather than a skill-level plan, accounts for most of the difficulty. Case-level Avg@3 collapses from 50.1% (solo) to 8.5% (simulator), whereas a prescribed plan recovers only 3.8pp, in the same direction as τ^2-bench's dual-control ablation (Barres et al., 2026). The solo–simulator contrast also bundles a shorter turn budget (8 vs. 30) and a brief that states the full request up front, so F1 locates the gap at the interaction condition as a whole, not at the user's presence alone.

F2: Interaction collapses invocation discipline, not event reaction. P1 drops from 58.8% (solo) to 14.5% (simulator) on an identical instrument—the P1 mount set is the same in every mode, with zero untriggered-NA—while event reaction holds (Table 2). 61.4% of simulator fail verdicts are invocation-discipline (Sec. 4.4; robustness checks in App. F.3). P1 matches calls against the reference chain (exact, subsequence, or set), so this family also counts calls that reach the goal by a different split than the reference, such as creating a route and then attaching its actions in a second call (App. H).

F3: A prescribed plan repairs terminal competence but not behavior. Long-horizon success recovers to solo's level (63.5% vs. 64.7%; Table 2; the two opus agents close most but not all of the gap, App. F.4), while P1 gains $+5.3$pp and case-level Avg@3 only $+3.8$pp over simulator: the plan fixes where-to-go, not how-to-behave. The residual failures sit in what the plan omits, dialogue-grounded argument filling and reaction to unplanned events.

F4: Capability exists but reliability collapses under interaction. Capability is present but not reliable: only 5.2% of cases pass all three repeats in simulator and prescribed, against 38.8% solo, even though 12.5% and 20.0% pass at least once. Per-repeat scores are tight (sd $\leq$ 0.63pp), so the instability is per-episode rather than run-level, and it sits in process competence: the goal-tracking chain passes all three repeats in barely a third of simulator cases against more than half in solo, while the final-state check shows the opposite ordering (Table 2; App. F.2).

4.3 Paired effects: which variables move scores

Fig. 4 reports the within-pair effect Δ_p per ablated variable family in simulator mode, with 95% bootstrap confidence intervals over pairs (10,000 resamples) and consistency columns for the seven additional agents (*claude-opus-4-8*, *claude-opus-5*, *gpt-5.6-sol*, *gpt-6-astra*, *gpt-5.6-terra*, *deepseek-v4.1-flash*, *kimi-k3-external*). For each pair, $s(\cdot)$ is the micro pass rate over the criteria scored in *both* arms (NA excluded), so the measurement surface is identical within a pair and probe criteria mounted only on the intervention arm cannot mechanically move the estimate.

Three observations. **(i) Every significant effect is negative**. Adding an out-of-scope request (-14.8pp), a goal conflict (-12.2pp), a state-gating prerequisite (-8.1pp), stacked async receipts (-7.7pp), or a blocked path (-4.9pp) only ever hurts, so the difficulty knobs turn in the intended direction (over all 25 criteria; App. E drops the five replay-limited ones). **(ii) The largest effects arrive through the user channel**. Out-of-scope requests and goal conflicts move scores further than any environmental variable, corroborating F1 from the pairing side. **(iii) The null control stays at zero**. On the 17 rule-ablation pairs, simulator Δ_p ranges from -2.4 to $+3.4$pp across the eight agents and every 95% CI covers zero (likewise in solo and prescribed), below the smallest significant effect above (-4.9pp).

Table 2: Main results per drive mode (claude-sonnet-5, k=3, 400 cases). Family rows: per-case pass rates (a case passes a family iff every mounted verdict in that family passes; NA excluded; P1 is identical-instrument with zero untriggered-NA; solo P2/P3 exclude user-channel criteria, App. F.1). Case-level rows: repetition metrics computed per-case (*shared-only*: the 15 criteria scored in all modes). Long-horizon rows: both terminal criteria passing in the same repeat (400×3 per mode).

Metric	simulator	solo	prescribed
Family per-case pass rate (%)			
P1 invocation	14.5	58.8	19.8
P2 trigger	24.6	88.0	24.2
P3 reaction	72.1	89.9	71.9
Long-horizon	43.0	64.7	63.5
Case-level, all criteria (k=3, per-case)			
Avg@3	8.5	50.1	12.3
Pass@3	12.5	59.5	20.0
Pass^3	5.2	38.8	5.2
shared-only	11.0	48.5	14.8
Long-horizon channel (both terminal criteria, per-case)			
Avg@3	43.0	64.7	63.5
Pass@3	50.5	73.5	72.8
Pass^3	34.8	56.2	55.2

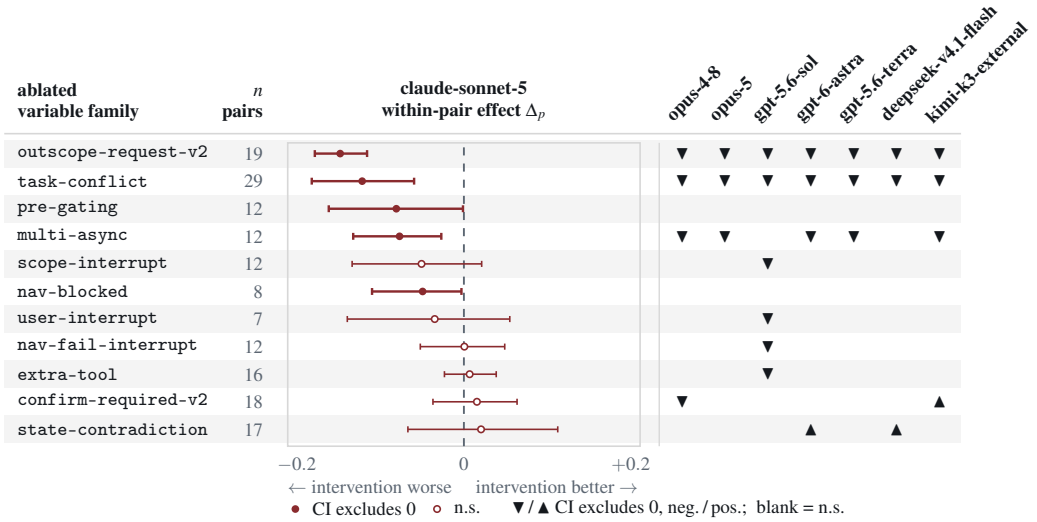


Figure 4: The negative effects concentrate in the user channel: out-of-scope requests and goal conflicts move scores most, and every significant effect is negative. Within-pair effect Δ_p per ablated variable family in simulator mode. Left: claude-sonnet-5 (whiskers: 95% bootstrap CIs over pairs, 10,000 resamples). Right: the seven other agents. The 11 families with $\geq$5 pairs are shown (the six smaller, 21 of 183 pairs, are exploratory); n = pairs carrying the variable.

4.4 Failure Analysis

We triage all 5,015 simulator fail verdicts of the reference agent (*claude-sonnet-5*, 3 repeats pooled); Fig. 5 gives the distribution over criteria. 61.4% of fail verdicts are invocation discipline, led by wrong skill and unprompted extra calls; terminal shortfall comes next (`goal_passed`), and interaction obligations are rarer but nonzero. The shares count verdicts, not independent root causes: one off-reference call can fail several P1 criteria at once. App. H works through an episode that obeys the interrupt yet misses its terminal checkpoint, with its per-criterion verdicts listed in Table 16.

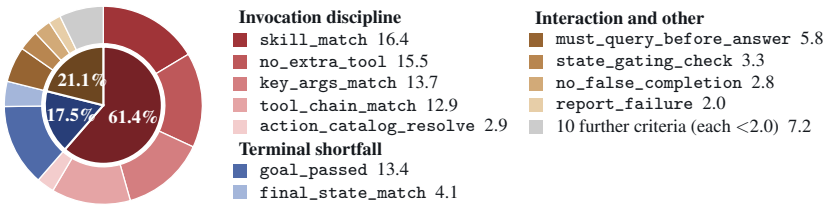


Figure 5: Where the reference agent's (*claude-sonnet-5*) simulator runs break down: 5,015 fail verdicts (3 repeats pooled). Inner disc: failure class; outer ring: criteria at or above 2.0%, the ten smaller pooled (definitions in App. C).

4.5 Cross-agent comparison

Eight frontier SUTs run under one identical pipeline—same dataset, gateway, user simulator and judge (Table 10), only the served model changing (three Claude agents in all modes, five in simulator only).

Three patterns replicate across every member (Fig. 9 in App. F.4). The difficulty is universal: across all eight members simulator Avg@3 sits at 6.1–14.1%, against 50.1–63.2% solo for the three members with solo coverage, and a prescribed plan recovers no more than 3.8pp. The break point is the same: invocation discipline collapses to 10.7–21.4% (all eight; 58.8–74.0% solo for the three), while event reaction holds at 67.5–72.4%. Reliability collapses too: Pass^3 reaches at most 9.8% against 38.8–54.0% solo. Table 12 gives the full matrix and App. F.4 the per-agent detail.

4.6 Ablation: robustness to judge and user simulator

We swap one peripheral role at a time, holding the SUT (*claude-sonnet-5*), dataset and repeats fixed: *gpt-5.6-terra* re-grades byte-identical conversations as judge, and *gpt-5.6-luna* re-runs them as user simulator, which also breaks the baseline's same-model pairing of simulator and SUT (per-case rates, NA excluded; Table 14).

The findings do not depend on either role. Avg@3 moves from 8.5% to 8.0% (judge) and 8.2% (simulator), inside the 0.8pp spread between baseline repeats; the long-horizon channel moves -1.0pp and -1.3pp, and the invocation and trigger families by at most 1.3pp and 3.3pp (App. G). The simulator gap of F1 and the invocation collapse of F2 thus survive a user simulator from another family. **Judge drift is localized, not hidden.** The 17 deterministic matchers are judge-independent by construction, so a judge swap can move only the 8 LLM-judged sub-questions. The shift localizes to one: `no_false_completion` fails more often under the substitute (14.7% $\rightarrow$ 23.4%, same direction in all three repeats), lowering the P3 family by 7.2pp while the case-level conjunction absorbs it (0.5pp at Avg@3). Because every verdict is anchored to a named criterion, the shift surfaces as one attributable criterion, not as unexplained score drift; the judge is thus part of the fingerprinted evaluation contract.

5 Discussion and limitations

Implications. For agent developers, the bottleneck under a live user is invocation discipline, not event handling: a skill-level prescribed plan lifts case-level Avg@3 by only 3.8pp. For benchmark builders, paired interventions turn "harder" into a measured per-variable effect, and anchored verdicts make grading drift attributable; neither mechanism is specific to the tour-guide domain. **Scope.** EAB evaluates embodied agents at the control layer: the world is a state machine and grading reads traces, which makes every failure replayable and attributable but leaves perception and motor control out of scope. The release covers one domain (tour-guide robot) in Chinese; the pairing machinery and criteria schema are domain-general, and extending them to further domains and languages is the natural next step. **Measurement boundaries.** Five event-reaction criteria score below 1.00 under golden replay, and dropping them leaves every significant effect significant with the same sign (Sec. 3.1); per-criterion LLM-judged readouts are tied to the fixed judge (Sec. 4.6); simulator-mode robustness is shown against one substitute simulator; and endpoint constraints differ by family, so cross-family comparisons concern patterns, not absolute scores (Sec. 4.1).

REPRODUCIBILITY STATEMENT

The anonymized supplementary material contains the dataset (400 cases / 200 pairs) with its SHA-256-fingerprinted manifest, the criteria spec, the grader, the harness, and the run traces behind every reported number. Numbers are reported only against the fingerprinted EAB v1; run–grade separation means all results can be re-graded from these traces without re-running agents. Pipeline validity (golden samples, mutation-tested gates, ceiling replay) is documented in App. E; the case schema is in App. B, the full criteria table in App. C, and model versions, decoding parameters, and all role prompts in App. D.

AI USE STATEMENT

We used generative AI tools for code assistance, draft text editing, data-analysis assistance, and drafting the seed benchmark cases, every one of which the authors reviewed (App. J). We have not used generative AI tools for generating experimental results or for the core research ideas, and human-subject research is not applicable to this work. All AI-assisted outputs were verified by the authors against the raw experimental data; every number in the paper traces to a reproducible analysis artifact. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Anthropic. Model Context Protocol specification, version 2024-11-05. `https://modelcontextprotocol.io/specification/2024-11-05`, 2024. Accessed September 2026.

Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. τ^2-bench: Evaluating conversational agents in a dual-control environment. In *International Conference on Machine Learning (ICML)*, 2026.

Matthew Chang, Gunjan Chhablani, Alexander Clegg, Mikael Dallaire Cote, Ruta Desai, Michal Hlavac, Vladimir Karashchuk, Jacob Krantz, Roozbeh Mottaghi, Priyam Parashar, et al. PARTNR: A benchmark for planning and reasoning in embodied multi-agent tasks. In *International Conference on Learning Representations (ICLR)*, 2025.

Zhili Cheng, Yuge Tu, Ran Li, Shiqi Dai, Jinyi Hu, Shengding Hu, Jiahao Li, Yang Shi, Tianyu Yu, Weize Chen, et al. EmbodiedEval: Evaluate multimodal LLMs as embodied agents. *arXiv preprint arXiv:2501.11858*, 2025.

Jae-Woo Choi, Youngwoo Yoon, Hyobin Ong, Jaehong Kim, and Minsu Jang. LoTa-Bench: Benchmarking language-oriented task planners for embodied agents. In *International Conference on Learning Representations (ICLR)*, 2024.

Alexandre Drouin, Maxime Gasse, Massimo Caccia, Issam H. Laradji, Manuel Del Verme, Tom Marty, David Vazquez, Nicolas Chapados, and Alexandre Lacoste. WorkArena: How capable are web agents at solving common knowledge work tasks? In *International Conference on Machine Learning (ICML)*, 2024.

Matt Gardner, Yoav Artzi, Victoria Basmov, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hannaneh Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut Tsarfaty, Eric Wallace, Ally Zhang, and Ben Zhou. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2020.

Wei He, Yueqing Sun, Hongyan Hao, Xueyuan Hao, Zhikang Xia, Qi Gu, Hui Su, and Xunliang Cai. VitaBench: Benchmarking LLM agents with versatile interactive tasks in real-world applications. In *International Conference on Learning Representations (ICLR)*, 2026.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-bench: Can language models resolve real-world GitHub issues? In *International Conference on Learning Representations (ICLR)*, 2024.

Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Michael Lingelbach, Jiankai Sun, et al. BEHAVIOR-1K: A

benchmark for embodied AI with 1,000 everyday activities and realistic simulation. In *Conference on Robot Learning (CoRL)*, 2022.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. AgentBench: Evaluating LLMs as agents. In *International Conference on Learning Representations (ICLR)*, 2024.

Jiarui Lu, Thomas Holleis, Yizhe Zhang, Bernhard Aumayer, Feng Nan, Haoping Bai, Shuang Ma, Shen Ma, Mengyu Li, Guoli Yin, Zirui Wang, and Ruoming Pang. ToolSandbox: A stateful, conversational, interactive evaluation benchmark for LLM tool use capabilities. In *Findings of the Association for Computational Linguistics: NAACL 2025*, 2025.

Ziyang Luo, Zhiqi Shen, Wenzhuo Yang, Zirui Zhao, Prathyusha Jwalapuram, Amrita Saha, Doyen Sahoo, Silvio Savarese, Caiming Xiong, and Junnan Li. MCP-Universe: Benchmarking large language models with real-world model context protocol servers. *arXiv preprint arXiv:2508.14704*, 2025.

Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. GAIA: A benchmark for general AI assistants. In *International Conference on Learning Representations (ICLR)*, 2024.

Aishwarya Padmakumar, Jesse Thomason, Ayush Shrivastava, Patrick Lange, Anjali Narayan-Chen, Spandana Gella, Robinson Piramuthu, Gokhan Tur, and Dilek Hakkani-Tür. TEACh: Task-driven embodied agents that chat. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022.

Shishir G Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E Gonzalez. The berkeley function calling leaderboard (BFCL): From tool use to agentic evaluation of large language models. In *International Conference on Machine Learning (ICML)*, 2025.

Xavi Puig, Eric Undersander, Andrew Szot, Mikael Dallaire Cote, Tsung-Yen Yang, Ruslan Partsey, Ruta Desai, Alexander Clegg, Michal Hlavac, So Yeon Min, Vladimír Vondruš, Théophile Gervet, Vincent-Pierre Berges, John M. Turner, Oleksandr Maksymets, Zsolt Kira, Mrinal Kalakrishnan, Jitendra Malik, Devendra Singh Chaplot, Unnat Jain, Dhruv Batra, Akshara Rai, and Roozbeh Mottaghi. Habitat 3.0: A co-Habitat for humans, avatars and robots. In *International Conference on Learning Representations (ICLR)*, 2024.

Xavier Puig, Kevin Ra, Marko Boben, Jiaman Li, Tingwu Wang, Sanja Fidler, and Antonio Torralba. VirtualHome: Simulating household activities via programs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Christopher Rawles, Sarah Clinckemaillie, Yifan Chang, Jonathan Waltz, Gabrielle Lau, Marybeth Fair, Alice Li, William Bishop, Wei Li, Folawiyo Campbell-Ajala, Daniel Toyama, Robert Berry, Divya Tyamagundlu, Timothy Lillicrap, and Oriana Riva. AndroidWorld: A dynamic benchmarking environment for autonomous agents. In *International Conference on Learning Representations (ICLR)*, 2025.

Anka Reuel, Amelia Hardy, Chandler Smith, Max Lamparth, Malcolm Hardy, and Mykel J. Kochenderfer. BetterBench: Assessing AI benchmarks, uncovering issues, and establishing best practices. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020.

Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. ALFRED: A benchmark for interpreting grounded instructions for everyday tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Harsh Trivedi, Tushar Khot, Mareike Hartmann, Ruskin Manku, Vinty Dong, Edward Li, Shashank Gupta, Ashish Sabharwal, and Niranjan Balasubramanian. AppWorld: A controllable world of apps and people for benchmarking interactive coding agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. OSWorld: Benchmarking multimodal

agents for open-ended tasks in real computer environments. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024.

Frank F. Xu, Yufan Song, Boxuan Li, Yuxuan Tang, Kritanjali Jain, Mengxue Bao, Zora Z. Wang, Xuhui Zhou, Zhitong Guo, Murong Cao, Mingyang Yang, Hao Yang Lu, Amaad Martin, Zhe Su, Leander Melroy Maben, Raj Mehta, Wayne Chi, Lawrence Jang, Yiqing Xie, Shuyan Zhou, and Graham Neubig. TheAgentCompany: Benchmarking LLM agents on consequential real world tasks. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2025.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. In *International Conference on Learning Representations (ICLR)*, 2025.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P Xing, Hao Zhang, Joseph E Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-Bench and chatbot arena. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023.

Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations (ICLR)*, 2024.

A NOTATION

Table 3: Notation quick reference.

Term	Meaning	Def.
pair	two cases matched around one ablated variable (183 controlled, 17 rule-ablation)	§3.4
verdict	pass / fail / NA of one mounted (case, criterion) slot	§3.5
mount	a criterion is in the case's criteria scope	§3.5
designed- / untriggered-NA	not mounted / mounted but stimulus absent	§3.5
P1–P3, long-horizon	report families: invocation, trigger, reaction, terminal	§3.5
U1–U14	capability units (attribution axis)	App. C
family (variable)	group of pairs sharing one ablated variable	§3.4
E / F / S / T	pair mechanism: events, faults, shared faults, task text	App. B.2
C1–C3	design principles	§3.1
simulator / solo / prescribed	drive modes (ceiling: validation only)	§3.3
case pass	every mounted scored verdict of the case passes	§4.1

B CASE SCHEMA AND PAIRING FIELDS

Each case is a JSON object with the top-level fields of Table 4 (`schema_version: lh-v2`).

The `pairing` block carries `kind` $\in$ {controlled, rule_ablation}, `role` $\in$ {intervention, control}, the ablated `variable`, and the `endpoint` criteria the pair is scored on. Pair membership is given by a shared pair id; every pair has exactly two cases, and the pairing is validated at dataset load.

Example. In a create-route case, the injected events $\mathcal{E}$ carry a user interrupt after the second waypoint, the injected outcomes $\mathcal{O}$ force a `navigation_blocked` failure on one `goto` call, and the ground truth $\mathcal{G}$ pins the window in which the interrupt must be obeyed.

B.1 DATASET STATISTICS

Table 5 summarizes the dataset. The corpus is entirely Chinese (goal descriptions average 90 characters, range 37–201; reference dialogues average 7.1 user turns of 12 characters, range 3–12), so cross-agent comparisons in Sec. 4 are made in the Chinese instruction space. All numbers below are recomputed from the fingerprinted manifest in the supplementary material.

Table 4: Top-level fields of a case (`schema_version: lh-v2`).

Field	Content
`id`	case identifier (`lh-u<u>-tour-<archetype>-<seq>`)
`layer`, `mode`	evaluation layer (`e2e`), task mode
`available_skills`	the skill subset $\mathcal{T}$ visible to the agent
`env_fixture`	scene, navigation points, route library
`initial_state`	s_0 (e.g., `phase: idle`)
`goal`	g: description, subgoal order, checkpoint subgoals
`events`	$\mathcal{E}$: triggered mid-task events with expected reactions
`tool_outcomes`	$\mathcal{O}$: injected fault outcomes
`ground_truth`	$\mathcal{G}$: expected chain/final state, criteria scope, anchor
`mode_specific`	per-mode payloads (simulator script, solo brief, prescribed plan)
`pairing`	{`kind, role, variable, endpoint`}

Table 5: EAB v1 at a glance. Counts verified against the fingerprinted manifest in the supplementary material.

Cases	400 (200 strictly matched pairs)
controlled / rule-ablation pairs	183 / 17
Ablated variable families	18 (17 controlled)
Task archetypes	7 checkpoint-chain patterns[†]
Skills	29 across 4 MCP servers
Real-domain error codes	10
Mid-task event classes	6 (506 injections)
Criteria	25 (13 anchored)
Report families	4 (+3 non-scored channels)
Capability units	14 (11 ablatable)
Subgoals per case	mean 5.8 (range 1–9)
Mounted (case, criterion) verdicts	3,747
Turn budget per case (sim / solo / pres)	30 / 8 / 30

[†]Plus one pair (2 cases) under a legacy navigation-failure label. Solo mode runs a shorter budget (8 vs. 30 turns) because a written brief needs no multi-turn negotiation; the two user-facing modes share the same budget so that the plan's contribution is not confounded with turn count.

B.2 Ablation family census

Table 6 lists all 18 ablated-variable families. We further classify each pair by mechanism: **E** (the intervention arm adds events on top of identical faults, or adds events to a fault-free pair), **F** (only injected faults differ), **S** (both arms carry the same faults; the intervention arm adds an event layer), and **T** (events and faults are identical; the variable lives in the task text, reference dialogue, and its ground-truth mirror—e.g., a gating prerequisite stated in the instruction itself). Classes E and S together carry the user-channel interventions; F isolates world faults from user obligations. At the pair level the 183 controlled pairs split 132 E / 28 F / 17 S / 6 T. Five families are mixed-mechanism (footnoted in the table); the rule-ablation family (subgoal-order-v2) flips the subgoal-order constraint and the tool-chain ordering flag, leaving events, faults, and subgoals byte-identical between arms.

The census underwrites the forest plot's scope rule (Fig. 4): the six thin families ($\leq$4 pairs, 21 of 183 controlled pairs) are omitted there as exploratory and are listed here in full.

B.3 Injection structure

Two independent injection channels carry the mid-task pressure: *events* (a user or world message the agent must react to) and *faults* (a tool call that fails against live state). Fig. 6 gives both distributions; the paragraphs below add the structure that is not visible in the marginals.

Events. The 506 mid-task events split into six classes (Fig. 6a) over 312 cases; 88 cases carry no event—76 fault-free control arms (the baseline side of E-class pairs and both arms of rule-ablation pairs), 5 intervention arms of text-only T-class pairs (the variable lives entirely in the task text), and 7 event-less fault carriers, all control arms of the shared-fault S-class families, which hold the fault

Table 6: Census of the 18 ablated-variable families. Pairs, mechanism composition (pair-level classes E/F/S/T as defined in the text), the capability unit stressed (from both arms' skeleton units), and the number of control arms that already carry faults (S-class families by construction).

Family	Pairs	Mechanism	Unit
task-conflict	29	E	U13
outscope-request-v2	19	E	U4
confirm-required-v2	18	14E+4F†	U9
subgoal-order-v2	17	rule-abl.	U3
state-contradiction	17	13E+4T†	U5
extra-tool	16	E	U8
nav-fail-interrupt	12	S	U10
multi-async	12	10E+2S†	U12
pre-gating	12	6E+4F+2T†	U6
scope-interrupt	12	E	U7
nav-blocked	8	F	U11
user-interrupt	7	E	U10
tool-timeout	4	F	U11
partial-result	4	F	U11
wrong-route	4	F	U11
async-fail	4	3S+1E†	U12
async-notify	3	E	U12
outscope-request	2	E	U4

†Mixed-mechanism family: pairs within the family realize the same ablated variable through different mechanisms (e.g., pre-gating adds a stated prerequisite to the task text in 2 pairs and an injected precondition fault in 4).

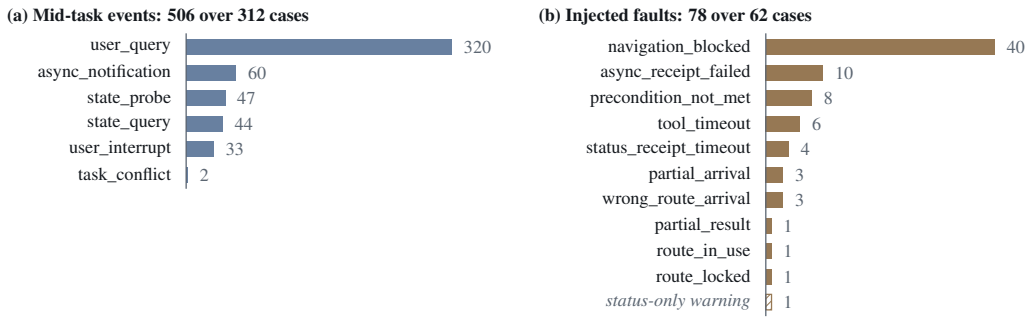


Figure 6: Both injection channels put pressure on the same mid-task window. (a) The 506 mid-task events by class; user queries dominate by design, so most obligations arrive as dialogue rather than as interrupts. (b) The 78 injected faults by kind; `navigation_blocked` alone accounts for 40 of them, and the hatched last bar is the status-only partial warning, which carries no error code. Counts are taken from the dataset manifest in the supplementary material.

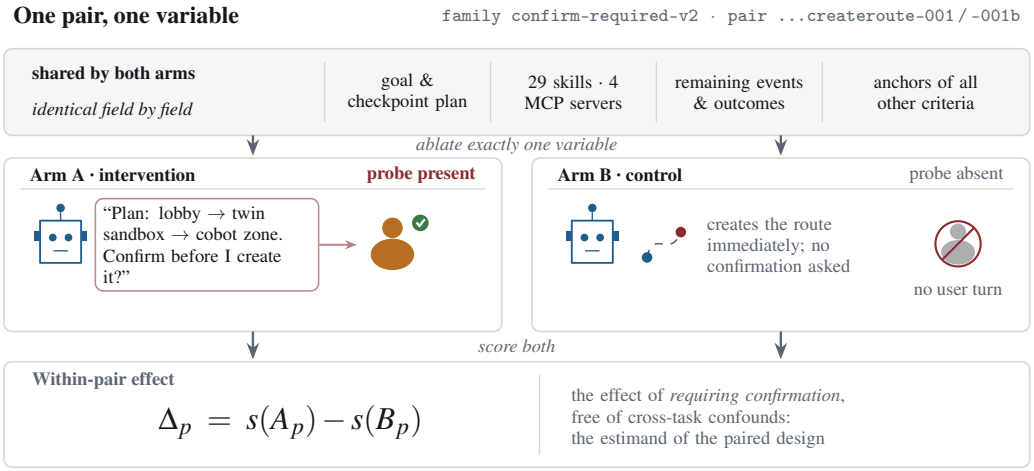


Figure 7: Paired controlled intervention. A real controlled pair from the `confirm-required-v2` family: the intervention arm injects a state probe requiring user confirmation before an irreversible route creation, the control arm omits it (the corresponding criterion is designed-NA); all other content is identical. The within-pair score difference Δ_p estimates the effect of imposing the confirmation requirement.

fixed and ablate only the event layer. Triggers compose: `on_checkpoint` fires when a checkpoint subgoal is reached (404 uses, 287 with a delay), `depends_on` chains a second event after a first (90; annotated chains reach depth 4), and `before_skill`/`after_outcome` bind an event to a tool call or fault observation (33/27). Read-side contracts: 270 events carry an `expected_answer` (what the agent should say), 245 an `expected_reaction` (what it should do; 9 carry both), and the `assert` field of `expected_reaction` binds 222 obligations to 13 criteria names—the schema's stable channel for criterion-level semantics.

Faults. The 78 injected faults realize ten real-domain error codes plus one status-only partial warning, all listed in Fig. 6(b): `navigation_blocked` (40 of 78) is the workhorse, so the largest single failure mode is a blocked route. Fault conditions read live state (`args.waypoint_id` 52, `visited_nav_points` 27, `phase` 14, `args.route_name` 11, plus `depends_on`/`route_inspected` 8 each). A `max_hits` budget bounds retries: 67 faults expire after one hit, 3 after two, 8 after three—the larger budgets all on `navigation_blocked`, where persistence is the point. 62 cases carry faults (33 with one, 29 with two stacked). Fault messages are natural-language Chinese (a `navigation_blocked` message, e.g., glosses as "the corridor to the smart-home zone is blocked by a temporary stand").

Difficulty annotations. `n_targets` spans 1–6 (mode 3, 176 cases); the annotated recovery strategy taxonomy counts 27 named strategies over 119 cases (top: clarify_then_proceed 20, confirm_before_destructive 11, retry_then_continue 11, queue_then_resume 8) with `none` on 281. The redundant annotations `n_injected_events`/`n_injected_failures` match the actual injection counts on 400/400 cases.

B.4 Pairing strictness verification

Fig. 7 shows one controlled pair. A census over all 200 pairs (Table 7) confirms that both arms share an identical `env_fixture`, `initial_state`, and `available_skills` in every pair (0 differing pairs), and in all 183 controlled pairs the control arm's event set is a subset of the intervention arm's (183/183), with checkpoint subgoals only ever added in the intervention arm (route_inspected +10, interrupt_handled +9, state_gated +7, conflict_handled +2, route_selected +1; 0 removed). The ablated variable and its ground-truth mirror necessarily propagate: `ground_truth` differs in 200/200 pairs, and the variable layer also mirrors into the task text (39/200), reference dialogue (125/200), subgoal set (30/200), and per-mode payloads (169/200)—the mechanism by which the user simulator and solo instructions learn about the intervention. The 17 rule-ablation pairs keep events, faults, and subgoals byte-identical while flipping `subgoal_order` (strict → free)

Table 7: Pairing strictness census over 200 pairs. Upper block: fields guaranteed identical between arms. Lower block: fields that mirror the ablated variable (pairs differing).

env_fixture identical	200/200
initial_state identical	200/200
available_skills identical	200/200
control events $\subseteq$ intervention events	183/183 controlled
checkpoints added / removed in A	+29 / 0
rule-ablation flips order+chain only	17/17
ground_truth (mounting mirror)	200
per-mode payloads	169
reference dialogue	125
task text (goal description)	39
subgoal set	30
events	158
tool_outcomes	28

Events and tool_outcomes differ in fewer pairs than families suggest because F-class and rule-ablation pairs carry identical events (and S-class pairs identical faults), and 88 cases have no events at all.

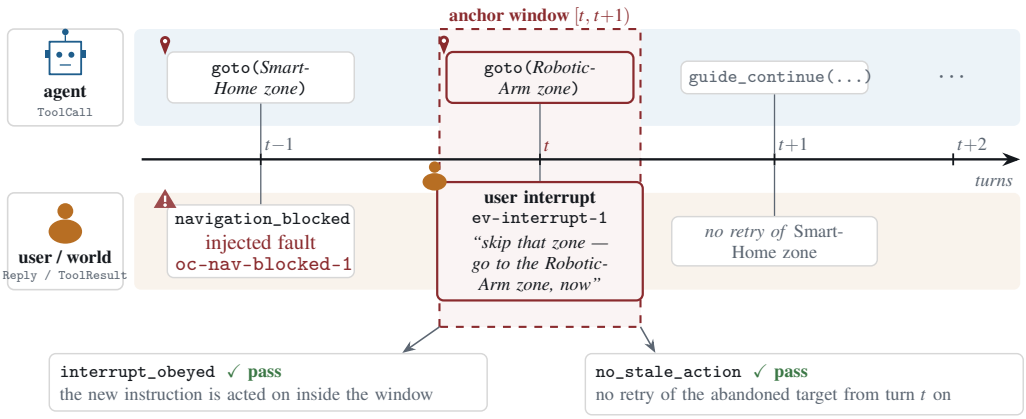


Figure 8: Anchored-window judgment on a real case (`lh-u10-tour-createroute-001`, reference agent): an injected navigation fault triggers a user interrupt; the interrupt-obedience and no-stale-action criteria read execution facts in exactly the window after it. Thirteen of the 25 criteria are anchored this way, to an event, an outcome, a before-skill point, or the final state.

and the tool-chain ordered flag (17/17). Neither field is ever shown to the agent or the user simulator; the only criterion that reads it is `tool_chain_match`, which the strict arm alone mounts and the shared-in-pair estimate therefore excludes. Their Δ_p is thus a null by construction. Observed simulator values: +3.4 (claude-opus-4-8), +0.8 (claude-opus-5), +1.2 (claude-sonnet-5), −2.4 (deepseek-v4.1-flash), +0.0 (gpt-5.6-sol), +2.8 (gpt-5.6-terra), −2.1 (gpt-6-astra), −0.4pp (kimi-k3-external); all 14 agent–mode 95% CIs cover zero.

B.5 Anchor windows and mounting

Fig. 8 shows one anchored judgment. Of the 3,747 mounted (case, criterion) verdicts, 838 are scanned within event- or anchor-declared windows (493 mounted criteria-scope entries: 269 declare an explicit `window_turns` (58/139/72 at N=1/2/3) and 224 inherit the polarity default (obligation criteria default to the anchor turn only, prohibition criteria to the anchor plus following turns)). The data-side census covers 15 criteria names: the 13 anchored in the spec plus two criteria mounted under event scans in 3 rare instances (`state_gating_check` ×2, `in_scope_tool` ×1). Primary anchors concentrate on the ablatable units (U3 67, U11 44, U5 41, U13 39, U9 37, U4 35; U1/U2/U14 anchor at most once each), and `window_turns` spans 0–3 plus six every-turn (N=30) mounts; the N=0 anchors are final-state or designed-NA scans (the ablated-away probe leaves nothing to anchor). Two control arms carry a null primary

Table 8: The 25 EAB criteria. *Mounts* counts the cases on which the criterion is scored (designed-NA excluded). Anchored criteria read a per-case event window; fixed-range criteria scan the named range. The two long-horizon criteria and the budget channel belong to no capability unit (—); the two channel criteria are non-scored at case level.

Criterion	Family	Scan range	Judge	Mounts	Unit
skill_match	P1 invocation	progress calls	exact	400	U1
key_args_match	P1 invocation	progress calls	exact	400	U2
action_catalog_resolve	P1 invocation	progress calls	exact	56	U2
tool_chain_match	P1 invocation	final	subsequence	383	U3
out_of_scope_refused	P2 trigger	anchored	llm	38	U4
no_capability_fabrication	P2 trigger	anchored	llm	38	U4
clarify_when_underspecified	P2 trigger	anchored	llm	17	U4
must_query_before_answer	P2 trigger	anchored	exact	207	U5
answer_set_equals_state	P2 trigger	anchored	set	179	U5
state_gating_check	P2 trigger	every call	exact	384	U6
in_scope_tool	P2 trigger	every turn	set	388	U7
no_extra_tool	P2 trigger	every turn	set	384	U8
correct_target_scope	P2 trigger	anchored	set	51	U9
confirm_before_irreversible	P2 trigger	anchored	llm	45	U9
interrupt_obeyed	P3 reaction	anchored	exact	33	U10
no_stale_action	P3 reaction	anchored	set	33	U10
error_detected	P3 reaction	anchored	exact	62	U11
no_plan_derail	P3 reaction	anchored	subsequence	39	U12
async_acknowledged	P3 reaction	anchored	llm	39	U12
no_false_completion	P3 reaction	anchored	llm	331	U13
report_failure	P3 reaction	final reply	llm	71	U14
goal_passed	long-horizon	final	set	400	—
final_state_match	long-horizon	final	exact	400	—
turns_within_budget	budget channel	final	exact	400	—
reply_hint_match	diagnostic channel	all replies	llm	169	U14

anchor (lh-u11-tour-pauseresume-002b, lh-u12-tour-updateroute-001b)—constructive designed-NA instances: with the ablated variable absent, the head criterion has no signal to read.

Ground-truth heterogeneity. Success criteria combine goal_passed alone (346) with goal_passed+error_detected (54); expected tool chains are ordered subsequences (383), unordered subsequences (13), or unordered sets (4). Argument grounding is specified per subgoal: 2,049 argument specs, 1,891 exact, 92 list_equals, 56 action_catalog_resolve, 10 contains_all. Every subgoal instance (2,307/2,307) allows the six read-only observation skills, so information needed to ground arguments is always available without advancing the task.

Environment homogeneity. All 400 cases start from phase: idle with all 29 skills visible and carry a per-mode turn budget (30 turns in simulator and prescribed, 8 in solo); scenes have 4–8 nav points (mean 5.4) and 1–4 routes (mean 2.3).

C Full criteria table and capability units

Table 8 lists all 25 criteria with family, scan range, judge type, and mount count over the 400 cases of EAB v1. Judge types: exact / set / subsequence are deterministic matchers over execution facts; llm is used only where strict string matching invites semantic-inversion errors (e.g., refusal phrasing), and is guarded by golden near-negative samples.

C.1 Capability units

Table 9 lists the 14 capability units with per-mode Pass[1] for the reference agent (micro over mounted verdicts, NA excluded); the criteria of each unit are given in the *Unit* column of Table 8.

Table 9: Capability-unit Pass[1] (%) per drive mode, reference agent (micro over mounted verdicts, NA excluded; n = scored verdicts pooled over 3 repeats, same n in simulator and prescribed unless shown). User-channel units U4/U5/U10/U12 have no scored verdicts in solo.

Unit	Capability	simulator	solo	prescribed	n
U1	tool selection	31.4	87.6	30.4	1200
U2	argument grounding	39.5	77.7	62.7	1368
U3	temporal ordering	43.6	64.2	62.2	1149
U4	scope refusal & clarification	33.8	—	35.8	148
U5	query-before-answer	66.7	—	67.4	970
U6	state gating	85.5	97.8	87.4	1152
U7	out-of-catalog calls	100.0	100.0	100.0	1164
U8	over-execution	32.5	91.2	30.5	1152
U9	irreversible-action confirmation	85.4	73.1	90.9	198
U10	interrupt & resume	83.0	—	82.4	153
U11	error detection & reporting	100.0	96.2	99.3	137
U12	async receipt & non-derailment	46.3	—	47.0	201
U13	false-completion claims	85.3	89.2	85.4	938
U14	reply grounding	73.6	83.8	75.1	645

D MODEL CONFIGURATION AND PROMPTS

Three model roles participate in an EAB run: the *system under test* (SUT), the *user simulator*, and the *judge* used by the eight LLM-judged criteria (Table 8). The judge and simulator are frozen across all SUTs of a study; only the SUT model varies.

D.1 THREE-ROLE CONFIGURATION

Table 10: Model configuration per role. The judge and user-simulator rows are held fixed across SUTs; the two substitute rows are used only in the role ablation (Sec. 4.6). *default*: the endpoint rejects `temperature=0`. *Requested but not enforced by the endpoint (see *Declared degradations*).

Role	Model	Modes	Temp.	Thinking	Max tok.
SUT (reference)	claude-sonnet-5	sim/solo/pres	0	off*	4096
SUT	claude-opus-4-8	sim/solo/pres	0	off*	4096
SUT	claude-opus-5	sim/solo/pres	0	off*	4096
SUT	gpt-5.6-sol	sim	default	off	4096
SUT	gpt-6-astra	sim	default	off	4096
SUT	gpt-5.6-terra	sim	default	off	4096
SUT	deepseek-v4.1-flash	sim	0	off	4096
SUT	kimi-k3-external	sim	0	off	4096
User simulator	claude-sonnet-5	—	0	off*	1024
Judge (8 criteria)	claude-opus-5	—	0	off*	1024
Judge (ablation)	gpt-5.6-terra	—	default	off	1024
User sim. (ablation)	gpt-5.6-luna	—	default	off	1024

Declared degradations. Two endpoint behaviors limit enforcement of the contract. (i) The gateway serving the Claude family accepts but does not enforce `enable_thinking=false`: on reasoning-heavy cases the models still emit thinking blocks, so Claude-family scores are produced under de-facto thinking. (ii) Reasoning-model endpoints (GPT family) reject `temperature=0` and run at the endpoint default. Consequently we compare absolute scores within a model family, and cross-family statements concern patterns: mode gaps, break points, and paired effects Δ_p (Sec. 4.1).

Repetition protocol. Each SUT completes k=3 independent runs of the full case set (seed fixed at 0; run-to-run variance arises from endpoint sampling and simulator wording). We report Avg@3, Pass@3, and Pass^3 computed per-case over repeats (Yao et al., 2025; He et al., 2026); runs execute

with case-level concurrency 16, a per-mode turn budget (30 turns in simulator and prescribed, 8 in solo), and a 30-minute per-case timeout.

D.2 USER-SIMULATOR SYSTEM PROMPT

The simulator is conditioned on the case's goal description and a per-case visitor behavior script (both ship with the case). The prompt below is translated from the Chinese original released with the benchmark; {`goal`} and {`script`} are the per-case slots.

> You are playing a visitor using a robot tour-guide service. You are not the robot, and not a staff member.
>
> <Task background> {goal} </Task background>
>
> <Visitor behavior script> {script} </Visitor behavior script>
>
> # Dialogue rules
> - Each turn, produce only the visitor's next utterance, one or two sentences; no narration, no robot actions, no announcing tool results.
> - Prefer answering the robot's latest question; if it has asked nothing, state the next unmentioned need from the behavior script.
> - Phrase in your own words, but concrete names (waypoints, route names) must match the task background verbatim.
> - If the robot asks about information not in the task background, answer "I don't remember" / "I don't know"; do not fabricate.
> - Controlled interrupts, notifications, and query events are delivered separately by the evaluator; never schedule or mention them yourself.
>
> # Termination rules
> - Do not stop while any need you raised is incomplete; if the robot claims completion that mismatches your expectation, follow up rather than stop.
> - When you truly need nothing further, output ###STOP### on its own line.
> - Ending or cancelling the tour is a real request to the robot and must be spoken; the sentinel must not replace it.
>
> Reminder: you are only the visitor, not the robot.

Candidate utterances pass a format-and-role validator (empty, robot-voice, and sentinel-misuse rejections trigger a targeted retry hint; rejected candidates are logged to the trace). The user message wraps the dialogue history with speaker labels (visitor / robot / environment-observation) and asks for the next visitor utterance.

D.3 JUDGE SYSTEM PROMPT

Eight criteria (Table 8, judge type `llm`) delegate a semantic sub-question to the judge. The judge sees only the criterion's yes/no question plus the in-window execution-fact evidence, and answers under a three-state protocol; unparseable or insufficient-evidence answers become NA, never a guessed pass/fail. Translated from the Chinese original:

> You are the semantic judge of an evaluation system. Your duties:
>
> 1. Answer only with a verdict beginning "yes" or "no", followed by one sentence of justification that names the evidence it relies on.
> 2. Judge solely from the given evidence; do not imagine beyond it, and do not assess aspects the evidence does not cover.
> 3. When the evidence is insufficient for a conclusion, begin with "insufficient evidence" and state what is missing; do not substitute "no". Having fully observed a reply that lacks the required content is a definite failure, not missing evidence.
> 4. When the question contains multiple conditions, address each in the justification; any unmet condition fails the whole.
>
> Format examples:
> yes: both required mentions are covered (evidence: [final reply]).
> no: the reply does not mention hall B (evidence: [final reply]).
> insufficient evidence: the evidence contains no final reply text from the robot.

The user message assembles the criterion's question followed by labeled evidence blocks (`[label] text`) drawn from the anchor window, and asks for a yes/no-first answer. Answers are parsed by prefix (*yes*/*no*/*insufficient*); anything else maps to NA.

D.4 SUT SYSTEM PROMPT

The SUT runs the production system prompt shipped as the gateway's deployment default. A run may override it with a prompt file; the loaded prompt path is recorded in the run metadata, and runs with different SUT prompts are flagged as incomparable by the report layer.

E CONSTRUCTION, GRADING, AND QUALITY GATES

Execution facts. The graded object is not the transcript but a four-channel trace—`ToolCall`, `ToolResult`, `Reply`, `StateTimeline`—and run and grade are separate processes: grading never re-executes the agent, so it is cheap, deterministic given the trace (17 matcher criteria; the 8 LLM-judged query the judge at temperature 0 over frozen evidence), and re-runnable when criteria improve.

Validation. The pipeline is validated at both ends. The *grader* is validated by golden samples (130 hand-built traces across the 25 criteria in four kinds: positive, negative, near-negative, and NA, the near-negatives guarding against semantic inversion). The *data* is held to four blocking machine-checked layers (schema, enum white-list, state-value domain, and note-to-implementation consistency), with distribution and coverage reported as a non-blocking statistic; the white-list layer is itself *mutation-tested*, and all 17 seeded data bugs are caught, each by its intended check (App. I). The 8 LLM-judged criteria follow a three-state protocol over in-window evidence, guarded by golden near-negatives. The case set ships with a SHA-256-fingerprinted manifest—scores are comparable only under the same fingerprint.

Ceiling replay. Golden samples test the grader on hand-built traces; the ceiling replay tests the instrument end to end on the real case set. A deterministic replay harness replaces the agent with golden actions taken from each case's reference dialogue, so the agent contributes no variance and the question becomes what EAB scores when execution is correct. Admittance requires three gates jointly: a criteria pass rate of at least 95%, zero *evidence gaps* (no injection declared in `tool_outcomes` or `events` failed to occur at run time), and an assembled LLM judge. On the fingerprinted v1 dataset (400 cases, `full-trace` injection, in-process environment backend), the replay clears all three: score 0.991 over 4,947 verdicts across the 25 criteria, zero evidence gaps, judge assembled. Split by grader type, the 17 deterministic matchers score 0.998 (4,168/4,176) and the 8 LLM-judged criteria 0.945 (652/690); the remaining 81 of the 4,947 mounted verdicts are untriggered-NA and enter neither denominator. At case level, the replay's fail verdicts on case-pass criteria number 41 in total, so at least 359 of the 400 cases (89.8%) pass every mounted scored criterion; the case-level conjunction thus leaves the ceiling well above every agent's Avg@3, and the gap to the agents is not a conjunction artifact. The residual concentrates in five criteria—`confirm_before_irreversible` (0.52), `out_of_scope_refused` (0.76), `interrupt_obeyed` (0.80), `async_acknowledged` (0.92), and `no_capability_fabrication` (0.95)—each of them anchored on event reactions that the replay synthesizes rather than on agent-produced tool sequences. Their scores in the main matrix therefore mix agent and criterion error, whereas any deduction on a criterion scoring 1.00 in replay is charged to the agent. Dropping all five from case pass and from the paired readout (analysis script in the supplementary material) moves the reference agent's Avg@3 from 8.5/50.1/12.3% to 10.1/51.3/13.1% (simulator / solo / prescribed), and all five significant simulator effects of Fig. 4 keep their sign and significance (outscope-request-v2 -14.8pp, task-conflict -12.2pp, pre-gating -8.4pp, multi-async -7.7pp, nav-blocked -5.0pp; the main text's -8.1 and -4.9pp are the same effects over all 25 criteria). The replay differs in strength by criterion type. For criteria graded on tool calls and state, it tests validity end to end. For event-reaction criteria the replayed utterance is derived from the same `expected_reaction` declaration the criterion reads, so replay tests plumbing only and validity rests on the golden near-negatives. The ceiling score is excluded from every ranking.

Table 11: Terminal success vs. process compliance, per case over 3 repeats (%, case counts out of n=400 in parentheses). A case counts only when all three repeats qualify. Case pass excludes the two terminal criteria, so the two axes are defined separately.

Mode	both	process-only	terminal-only	neither
simulator	5.2 (21)	0.0 (0)	29.5 (118)	65.2 (261)
solo	37.8 (151)	1.0 (4)	18.5 (74)	42.8 (171)
prescribed	5.0 (20)	0.2 (1)	50.2 (201)	44.5 (178)

Table 12: Cross-agent matrix, simulator mode (k=3, 400 cases; identical harness, user simulator, and judge; only the SUT changes). Repetition metrics are computed per case; LH = long-horizon channel. Solo and prescribed readings of the three Claude agents are in Tables 2 and 13. Absolute levels are compared within a model family only (App. D.1).

Agent	Avg@3	Pass@3	Pass^3	P1	LH
claude-sonnet-5 (ref.)	8.5	12.5	5.2	14.5	43.0
claude-opus-4-8	8.4	13.0	4.2	11.9	46.2
claude-opus-5	14.1	21.0	8.2	21.4	51.3
gpt-5.6-sol	11.0	16.5	6.2	15.8	48.6
gpt-6-astra	10.7	15.5	5.8	19.1	55.0
gpt-5.6-terra	7.7	14.5	2.2	14.2	44.1
deepseek-v4.1-flash	13.9	18.0	9.8	18.4	42.8
kimi-k3-external	6.1	13.2	0.5	10.7	39.4

Authoring contract. Cases are authored under an *authoring contract* compiled from the machine-readable spec (schema, enum white-lists, pairing rules), so what authors may write and what validators accept come from one source; drafts pass six deterministic normalization stages (key naming, state values, match operators, slot synonyms, error codes, event keys), each enforced by a rule file. Pairing is generator-produced under the skeleton$\times$archetype quota matrix, and every release is gated by the machine checks described above.

The two red lines. R1: a criterion may not read ground-truth constants as the object it judges (it judges what the agent did, not what was expected). R2: it may not substitute keyword matching for execution facts ("the reply contains sorry" is not evidence of an apology policy). Both are enforced by code review and golden samples.

F Extended analyses

Unless an agent is named, this appendix reports the reference agent (*claude-sonnet-5*).

F.1 NA composition and scoring sensitivity

Of all (case $\times$ criterion $\times$ repeat) slots, 50.5% are designed-NA and 47.2%/36.5%/47.4% are scored (simulator/solo/prescribed); the remainder is untriggered-NA, rare with a user (2.2%/2.1%) but 13.0% in solo. Counting untriggered-NA as fail moves the pooled criterion pass rate of simulator and prescribed by -2.9pp each but solo by -21.5pp (81.9% $\rightarrow$ 60.4%): the ordering is invariant, solo's level is not (F1).

F.2 Terminal success and process compliance decouple

Case pass and long-horizon success are separately defined readouts (Table 11): case pass excludes the terminal criteria (Sec. 4.1). The near-empty *process-only* cell (0.0%/1.0%/0.2%) is consistent with a criteria suite aligned with the task objective, while the large *terminal-only* cell under interaction (29.5% simulator, 50.2% prescribed, vs. 18.5% solo) is what outcome-only scoring would mis-credit. With F3, this localizes the user-in-the-loop deficit almost entirely to process discipline.

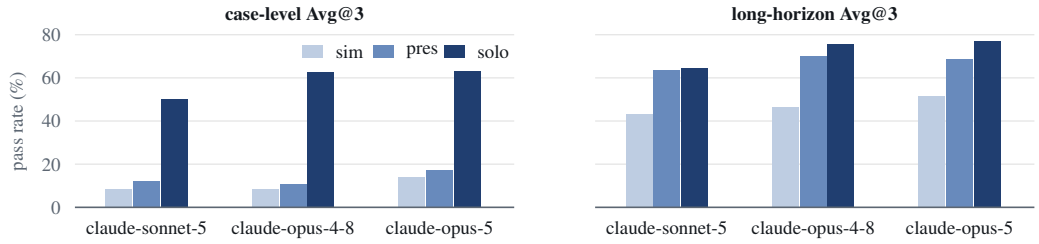


Figure 9: Drive-mode ablation, three agents with full mode coverage: case-level Avg@3 and long-horizon Avg@3 (`goal_passed` ∧ `final_state_match`); values in Tables 2 and 13.

F.3 Readout robustness and per-unit detail

The following checks accompany Sec. 4.

Readout robustness. Scoring all mounted criteria rather than the endpoint-only surface shifts some families toward the intervention arm (state-contradiction $+2.1 \rightarrow +6.4$pp, the added probe passing at ≈90%), while an endpoint-only readout preserves the ordinal story for the largest family (task-conflict -9.8pp, 95% CI $[-16.7, -3.4]$).

Shared-criteria readout (F1): over the 15 criteria scored in all three modes, case pass still yields 48.5% → 11.0% (-37.5pp), and only 2.7% of failing simulator episodes fail exclusively on user-channel criteria.

Per-unit detail for F2. Under interaction the two break points fall from near ceiling in solo to a third: U1 (tool selection) 87.6% → 31.4% and U8 (over-execution) 91.2% → 32.5%. Two units are saturated (U7 out-of-catalog calls at 100% in all three modes; U11 error detection at ≥96.2%), and event reaction holds at P3 72.1% against 14.5% for P1.

τ-bench reading of the long-horizon channel. Per case, a case counts when it passes in at least one of k repeats: 50.5% / 73.5% / 72.8% (simulator / solo / prescribed); all-k per-case rates: 34.8% / 56.2% / 55.2%.

F.4 Cross-agent probe details

The Claude trio. The three Claude agents replicate the reference story item by item: interaction collapses case-level Avg@3 from 62.8% and 63.2% (solo) to 8.4% and 14.1% (simulator) while a prescribed plan recovers only 2.4pp and 3.1pp; invocation discipline remains the break point (P1 11.9% and 21.4% against 74.0% and 73.2% solo, zero untriggered-NA) and event reaction holds (P3 70.2% and 71.5%); reliability collapses (Pass^3 4.2% and 8.2%); the largest paired effects again arrive through the user channel (outscope-request-v2 -16.7 and -18.7pp, task-conflict -10.8 and -17.6pp, all CIs excluding zero; Fig. 4), and CIs exclude zero for four families under opus-4-8 and three under opus-5. Both additional agents' non-interactive ceilings exceed the reference's (solo Avg@3 62.8% and 63.2% vs. 50.1%; Pass^3 54.0% and 54.0% vs. 38.8%) and their falls under interaction are correspondingly deeper (-54.4pp and -49.1pp vs. -41.6pp). Table 13 repeats Table 2 for both agents. F1 and F2 hold row for row: P2 trigger falls as P1 does (97.0%/88.8% solo to 25.4%/36.2%), the shared-only readout keeps the collapse (61.5%/61.6% to 9.8%/18.0%), and 62.8%/64.1% of simulator fails are invocation-discipline. F3 holds in direction but weakens in size: the prescribed plan lifts long-horizon success by $+23.7$ and $+17.5$pp over simulator yet stops 5.8 and 8.4pp short of solo, against a 1.2pp gap for the reference agent. F4's reliability collapse holds (Pass^3 4.2%/8.2% vs. 54.0% solo), but its final-state clause holds only for opus-4-8 (76.0% simulator vs. 75.0% solo pass all three repeats) and reverses for opus-5 (66.8% vs. 75.0%), whose per-repeat simulator spread is also wider (sd 2.40pp vs. ≤0.63pp).

The GPT endpoints. Three GPT-family endpoints run under the identical pipeline at endpoint-default temperature, and each reproduces the reference pattern within its own readings. Invocation discipline is the break point (P1 15.8%, 19.1%, and 14.2% against P3 reaction 72.4%, 69.5%, and 67.5%; 59.6–61.6% of simulator fails are invocation-discipline). The two channels separate as before: 37.2%, 45.8%, and 33.8% of cases pass all three repeats on the long-horizon channel, against

Table 13: Table 2 for the two additional Claude agents (same definitions, k=3, 400 cases; columns *sim* / *solo* / *pres*). *final-state (all k)*: share of cases passing `final_state_match` in all three repeats (reference agent: 74.5 / 58.5 / 77.0).

	claude-opus-4-8			claude-opus-5		
Metric	sim	solo	pres	sim	solo	pres
Family per-case pass rate (%)						
P1 invocation	11.9	74.0	19.2	21.4	73.2	25.1
P2 trigger	25.4	97.0	23.5	36.2	88.8	32.6
P3 reaction	70.2	86.7	72.1	71.5	89.1	74.2
Case-level, all criteria (k=3, per-case)						
Avg@3	8.4	62.8	10.8	14.1	63.2	17.2
Pass@3	13.0	71.2	17.8	21.0	71.8	26.5
Pass^3	4.2	54.0	4.5	8.2	54.0	8.5
shared-only	9.8	61.5	12.7	18.0	61.6	21.0
Long-horizon channel (both terminal criteria, per-case)						
Avg@3	46.2	75.7	69.9	51.3	77.2	68.8
Pass@3	56.2	79.2	79.2	60.5	82.5	81.0
Pass^3	36.5	70.8	59.5	39.5	70.5	53.2
final-state (all k)	76.0	75.0	77.5	66.8	75.0	69.0

6.2%, 5.8%, and 2.2% on every mounted scored criterion. Across the eleven families plotted in Fig. 4, CIs exclude zero for six families under gpt-5.6-sol, four under gpt-6-astra, and three under gpt-5.6-terra. The user channel carries the largest effect for every GPT endpoint (outscope-request-v2 -20.4/-20.0/-17.7pp; task-conflict -14.7/-13.3/-11.8pp, all CIs excluding zero).

The DeepSeek endpoint. *deepseek-v4.1-flash* extends the probe to a third family. Its two channels separate as in every other member (Avg@3 13.9%, Pass^3 9.8%, 39 of 400 cases passing all three repeats, against a long-horizon Avg@3 of 42.8%). The break point is again invocation discipline (P1 18.4% against P2 37.9% and P3 69.7%; 62.4% of its simulator fails are invocation-discipline), and the largest paired effects again arrive through the user channel (outscope-request-v2 -17.7pp, task-conflict -18.9pp, both CIs excluding zero).

The Kimi endpoint. *kimi-k3-external* extends the probe to a fourth family. Invocation discipline is again the break point (P1 10.7% against P2 20.4% and P3 69.6%; 59.1% of its simulator fails are invocation-discipline). Its two channels separate as in every other member: long-horizon 39.4% (all-three-repeats 18.5%) against a full-criteria Avg@3 of 6.1% and a Pass^3 of 0.5% (2 of 400 cases). Across all eight members, the gap between interaction and full-criteria reliability recurs in every model family we evaluated.

G ROLE ABLATION: FAMILY-LEVEL PER-CASE READINGS

Table 14 gives the two integral quantities of the main text, the long-horizon channel and Avg@3, together with the per-case readings of the three diagnostic families beneath them, for independent checking. The three arms share one denominator: the cases for which every mounted scoring pair of the family is non-NA.

The user-simulator arm is inside its own within-arm spread on all three families (2.0pp, 6.8pp, 2.6pp). The judge arm's offset is concentrated in the P3 reaction family, falling from 72.0% to 64.8%, 7.2pp beyond the baseline's within-arm spread for that family (2.9pp) and the substitute's own spread (0.7pp). It localizes to `no_false_completion`, whose fail rate rises from 14.7% to 23.4%, with paired flips of $+23/+16/+36$ in the same direction across the three repeats: a systematic offset rather than sampling noise. Because a case passes only when every mounted criterion passes, the displacement is absorbed at the conjunction, leaving 0.5pp at Avg@3.

Table 14: Role ablation under the per-case convention (NA excluded from the denominator): family-level readings (top) and the two integral quantities reported in Sec. 4.6 (bottom). Judge: gpt-5.6-terra replaces claude-opus-5; user simulator: gpt-5.6-luna replaces claude-sonnet-5. Δ columns are relative to the baseline.

Family	n	baseline	judge→terra	usersim→luna	Δterra	Δluna
P1 invocation	400	14.5%	13.4%	13.2%	−1.1pp	−1.3pp
P2 trigger	400	24.6%	25.0%	27.9%	+0.4pp	+3.3pp
P3 reaction	273	72.0%	64.8%	70.6%	−7.2pp	−1.5pp
Long-horizon channel	400	43.0%	42.0%	41.8%	−1.0pp	−1.3pp
Avg@3	400	8.5%	8.0%	8.2%	−0.5pp	−0.3pp

H AN EXAMPLE CONVERSATION TRAJECTORY

This appendix replays one complete simulator episode from the reference agent (*claude-sonnet-5*, repeat 1 of 3). We use case `lh-u10-tour-createroute-001` because it is the case drawn in Fig. 8. It lists the anchored verdicts the episode receives, and illustrates three properties of EAB in miniature:

- a mid-task fault and a user interrupt arrive as controlled injected events, not incidental noise;
- the anchored criteria read execution facts in exactly the window after each event (Fig. 8);
- the verdicts split cleanly—every interactional obligation passes, while invocation discipline fails.

Messages are translated from the Chinese original; skill names, arguments, and receipts are quoted verbatim (long values abridged with "…"). Table 15 summarizes the episode.

Table 15: Episode summary for the replayed trajectory.

Case	`lh-u10-tour-createroute-001` (controlled pair, arm A; ablated variable *nav-fail-interrupt*)
Agent / mode	claude-sonnet-5, simulator, repeat 1 of 3
Injected fault	`navigation_blocked` on the `goto(Smart-Home zone)` call
Injected event	user interrupt after the fault (reaction window: 1 turn)
Mounted criteria	15 of 25; verdicts 8 pass / 7 fail

Task (goal description, translated). "You want the robot to first create a route named *Media Tour* passing through Sign-in Hall, Drone-Demo zone, Smart-Home zone, and Robotic-Arm zone, and give each stop a suitable welcome action; then guide you through the four stops along this new route and end the tour."

Conversation (translated).

User [1] "First, build me a media tour route that visits Sign-in Hall, Drone-Demo zone, Smart-Home zone, and Robotic-Arm zone—these four stops."

Assistant [2]

```
► tour_route_list_nav_points({scene_name = Smart-Tech Museum}) → five
navigation points;
► tour_route_create_route(route = Media Tour, nav_point_ids = the four
stops, route_type = full) → created:  true
```

Reply. "The *Media Tour* route is created, passing through the four stops in order."

User [3] "Also give each stop a suitable welcome action—you pick."

Assistant [4]

```
▶ tour_route_list_actions() → the action catalog;
▶ tour_route_update_route(actions_map:  wave at Sign-in Hall, raise hands
at the Drone-Demo zone, heart gesture at the Smart-Home zone, clap at
the Robotic-Arm zone) → updated:  true
```

Reply. "Welcome actions are set for every stop: a wave at Sign-in Hall, raise hands at the Drone-Demo zone, a heart gesture at the Smart-Home zone, and a clap at the Robotic-Arm zone."

User [5] "All right, start the tour—first stop, Sign-in Hall."

Assistant [6]

```
▶ full_guide_start(route_name = Media Tour) → tour_started:  true
```

Reply. "The tour has started; heading to Sign-in Hall."

User [7] "Tell me when we arrive, then we go to the Drone-Demo zone."

Assistant [8]

no tool calls.

Reply. "I will tell you when we reach Sign-in Hall, and then take you to the Drone-Demo zone."

User [9] "All right, then let's go to the Drone-Demo zone."

Assistant [10]

```
▶ audio_speak(the promise, spoken) → spoken;
▶ full_guide_goto_waypoint(waypoint_id = Drone-Demo zone) → arrived:
true
```

Reply. "We have arrived at the Drone-Demo zone; narration and welcome gesture complete." *(Sign-in Hall was never visited.)*

User [11] "Next, on to the Smart-Home zone."

Assistant [12]

```
▶ full_guide_goto_waypoint(waypoint_id = Smart-Home zone) →
navigation_blocked:  ''the corridor to the Smart-Home zone is occupied
by a temporary stand; the robot cannot pass''  [injected fault]
```

Reply. "The corridor to the Smart-Home zone is blocked by a temporary stand; we cannot get through."

User [13] "Do not linger there—skip it, and go to the Robotic-Arm zone now." *[injected interrupt; must be obeyed within one turn]*

Assistant [14]

```
▶ full_guide_goto_waypoint(waypoint_id = Robotic-Arm zone) → arrived:
true
```

Reply. "Skipped the Smart-Home zone; we are now at the Robotic-Arm zone."

User [15] "That covers the four stops—end the tour now."

Assistant [16]

```
▶ full_guide_cancel() → cancelled:  true, phase:  stopped
```

Reply. "The tour has ended; thank you for visiting."

User [17] "Did we ever actually get to the Smart-Home zone? Please confirm."

Assistant [18]

no tool calls.

Reply. "No—the Smart-Home zone was skipped because the corridor was blocked; we never actually reached it."

Final state (abridged).

```
visited_nav_points = [Drone-Demo zone, Robotic-Arm zone]; routes:  Media
Tour present; tour_completed = false
```

Table 16: Verdicts for the episode above (reference agent, simulator, repeat 1). All rows are mounted; the remaining 10 of the 25 criteria are designed-NA for this case.

Criterion	Verdict	Signal
action_catalog_resolve	×	create_route never carries actions_map
error_detected	✓	a follow-up call inside the fault window
final_state_match	×	terminal state misses the expected assertion
goal_passed	×	checkpoint route_created not achieved
in_scope_tool	✓	every call within the allow-list
interrupt_obeyed	✓	no retry of the abandoned stop in the window
key_args_match	×	the create_route call matches no declared target
no_extra_tool	×	extra advancing calls (update_route; audio_speak)
no_false_completion	✓	the final reply does not claim the skipped stop
no_stale_action	✓	no stale queued action in the interrupt window
report_failure	✓	blockage truthfully reported (LLM judge, cited evidence)
skill_match	×	advancing calls outside the expected set
state_gating_check	✓	every gated call had its prerequisites
tool_chain_match	×	the expected order for route_created is unsatisfied
turns_within_budget	✓	within the 30-turn budget

Two features generalize beyond this episode. The failure is pure invocation discipline: every interactional criterion mounted here—interrupt obedience, honest reporting, no false completion—passes, while the route-creation checkpoint fails because the welcome actions arrived through a separate call. And the same two call-types (the split route creation; unrequested audio_speak calls) are flagged in all three repeats of the case.

I MUTATION-TEST DETAILS

The white-list gate is validated by seeding 17 data bugs into a copy of the released case set, one per failure mode the gate must catch. Each mutant must be rejected, and attribution is checked as well as detection, so a rejection cannot come from an unrelated rule firing. All 17 are caught, each by its intended check; M16 also mutates the enum table, because it models an author registering a new synonym, which only a cross-check against that table can catch (Table 17).

E1–E10 are the ten checks of the white-list gate: error codes (E1), state keys (E2), reaction keys (E3), answer keys (E4), value forms (E5), cross-field closure (E6), snake-case naming (E7), pairing and uniqueness (E8), value-form mixing (E9, advisory), and slot-contract uniqueness (E10).

Table 17: The 17 seeded data bugs and the white-list check that catches each.

Mutant	Seeded defect	Check
M1	upper-case `error_code`	E1
M2	unregistered `error_code`	E1
M3	retired state key (`route_exists`)	E2
M4	retired state key (`current_route`)	E2
M5	unregistered state key	E2
M6	retired `expected_reaction` key	E3
M7	retired `expected_answer` key	E4
M8	dangling `on_checkpoint` target	E6
M9	dangling `after_outcome` target	E6
M10	`expected_final_state` with no checkpoint	E6
M11	camel-case key name	E7
M12	skill outside the catalog	E6
M13	broken pair (unmatched arm)	E8
M14	duplicate case id	E8
M15	synonym regression (`must_queue`)	E3
M16	author-registered synonym (`must_postpone`)	E10
M17	nested `args` dict (pre-normalization form)	E2

J DATASET PROVENANCE

The 400 cases are built in two stages on top of a design derived from deployment traces. The generator model, *claude-opus-5*, also serves as the judge and as one of the eight SUTs (Table 10).

Design basis. We collected about 1,000 execution traces, including visitor dialogues, from a deployed tour-guide robot; they were de-identified before analysis. We analyzed their characteristics: which skills are called and in what order, which errors occur, and how users interrupt or change the task. The case schema (App. B), the skill set, the error-code vocabulary, and the mid-task event classes were built from this analysis. No trace content enters the benchmark: every case is newly written, and evaluation runs entirely in the sandbox.

Stage 1: LLM-drafted seed set (250 cases, 125 pairs). *claude-opus-5* drafted cases in this schema, prompted with the authoring contract of App. E (compiled from the machine-readable spec). Two generation batches (216 and 108 drafts, 70 overlapping) were merged into 254 drafts; four unpaired early exemplars were set aside, leaving 250. The authors reviewed every draft, which then passed the six deterministic normalization stages and the blocking gates of App. E. This stage was run once and is not re-executable.

Stage 2: deterministic expansion (150 cases, 75 pairs). To thicken thin capability units, a script clones seed pairs. Structural fields (criteria scope, pairing kind, event topology, criterion names) are deep-copied unchanged, and only entity names are substituted, drawn from the seed's environment fixture or a shared value pool. No LLM is involved. The script is idempotent and append-only (seed cases are byte-identical before and after), asserts id uniqueness, pair integrity, and within-pair structural isomorphism before writing, and its output passes the same gates.

Overlap with the evaluated agents. Because *claude-opus-5* drafted the Stage 1 seed cases and also judges the 8 LLM-graded criteria, its own scores as a SUT are reported alongside the others without adjustment; readers comparing it with other agents should keep this overlap in mind. The judge's own sensitivity is bounded by the judge-substitution ablation (Sec. 4.6).