StoryHARN: Handling Execution Conflicts in Interactive Stories

Anonymous authors
Paper under double-blind review

Abstract

Language-model storytellers can receive explicit world facts and rules yet propose state changes that violate them. We study this execution conflict: a candidate update is incompatible with the supplied contract, regardless of whether the model can recall its text. We introduce StoryHARN, a runtime layer that checks updates before commitment and handles conflicts through rule-compatible alternatives, isolated what-if branches, and deferred intentions requiring later confirmation. Decisions are recorded for inspection and replay. Across five configurations, 1,200 turns and 12 authored worlds, we separate structured-state validity from action coverage and a continuation proxy. In the initial comparison, StoryHARN achieves 67.5% state validity, versus 39.2% for rule prompting and 90.8% for hard rejection. Trace analysis exposes an incomplete admission check and an overly restrictive evidence rule. A separately versioned development follow-up commits 1,020 of 1,200 planned turns; both monitored configurations have no committed-state violations, but no continuation advantage over rejection is established. A separate 200-rating study of 100 three-turn traces illustrates narrative evaluation beyond state validity, not repair efficacy. The contribution is an inspectable execution interface and failure analysis under explicit contracts; improved memory, latent rule understanding, and player-intent fidelity are not established.

1 Introduction

A narrator receives the facts that an archive is locked and the player lacks its key, together with a rule requiring the key for entry. It nevertheless proposes moving the player inside. The immediate failure is an incompatible state transition despite supplied constraints. We call this an execution conflict.

Our focus is the boundary between proposing and committing an action. "Available facts" means that relevant state and rules are present in the narrator input, not that we observe the model's internal understanding. Admission checks prevent conflicting candidates from silently becoming canonical state; missing-fact retrieval is outside our scope.

StoryHARN checks typed updates against executable rules and records three declared treatments: translation, what-if branching, and deferral. The policy is deterministic and inspectable; it neither learns intent nor supplies a long-term memory system. A branch scopes an exception without rewriting the mainline. Authorized edits to the contract itself are outside the evaluated protocol.

Our five-configuration experiment uses 12 authored worlds and twenty turns per episode. Three conditions receive explicit rules; the other two are contextual controls (Section 4). Hard rejection has higher initial state validity than StoryHARN, while their continuation-proxy difference is uncertain. Trace inspection reveals a monitor that misses changed rule dependencies, compounded by a contract requiring exactly three pieces of evidence after unlocking an ending.

Our contributions are threefold: an executable interface for handling conflicting updates beyond accept-or-reject; a versioned set of 240 authored actions separating public inputs

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.

from evaluation-only effects; and a five-configuration study diagnosing the gap between local admission and full-state validity. The evidence supports an auditable execution interface and failure analysis, not better memory, enjoyment or player-intent fidelity.

2 Problem Setting and Related Work

Interactive narrative. Interactive storytelling has long balanced player choice and authored coherence (Riedl & Bulitko, 2013; Mateas & Stern, 2003). Language models broaden the action space and make causal commitments easier to change implicitly. We study the transition from attempted action to recorded state, rather than replacing plot design or dialogue generation.

Established approaches to intervention. Narrative mediation distinguishes accommodation, which replans around an accepted exceptional action, from intervention, which substitutes an action to protect causal structure (Riedl et al., 2003). Shielding monitors proposed actions and corrects unsafe ones relative to a formal specification (Alshiekh et al., 2018). Checking or replacing a conflicting action is therefore not itself a new contribution. Our focus is the explicit state interface for three repair dispositions and an auditable narrative stress test; we do not synthesize a temporal-logic shield.

State and model separation. Statler separates world-state reading from writing (Yoneda et al., 2024), while LLM-Modulo combines model suggestions with external model-based verification (Kambhampati et al., 2024). Our monitor applies this separation to execution conflicts in narrative transitions. Supplying a rule and enforcing the resulting state transition are distinct interventions. Memory and feedback offer other ways to support behavior over time (Park et al., 2023; Shinn et al., 2023). Our history and rule-prompt controls are simpler interventions, not implementations of these complete systems; serializing rules performs no graph retrieval or learning.

Commitment evaluation. Narrative Commitment Preservation (NCP) evaluates commitments and progression in interactive narratives (Ma et al., 2026). Its separation between the narrator and evaluation motivates our information boundary. NCP also permits a blocked or redirected request when it is acknowledged and explained; accepting every player request is not its success criterion. Our study asks a narrower operational question: after a candidate update conflicts, what state and continuation does a repair leave behind? Our original worlds have an NCP-compatible specification view, but this experiment does not run the official NCP evaluator or its dynamic player protocol. Compatibility of file layouts is not an NCP result, and our scores are not directly comparable to its reported scores.

Three distinct targets. We distinguish structured-state validity, prose-to-state consistency, and fidelity to a human player's intent. Our execution claim concerns only the first, checked by our small rule language. An empty update can be valid while its prose is misleading; a declared alternative can be valid but unwanted. These differences govern our interpretation of every score.

3 The StoryHARN Runtime

3.1 States, candidates, and contracts

Let s_t be the canonical state before turn t, a_t a player's natural-language action, and h_t the public history. The narrator returns a proposed paragraph y_t, a short intent description z_t, and a partial state update Δ_t. We write $s_t \oplus \Delta_t$ for overwriting the specified fields while keeping all other fields unchanged. The current reference state schema has 14 named fields, including location, player role, access permission, evidence count, and ending flags. Closed-schema validation rejects undeclared fields and incompatible types; it does not assess whether a paragraph is true.

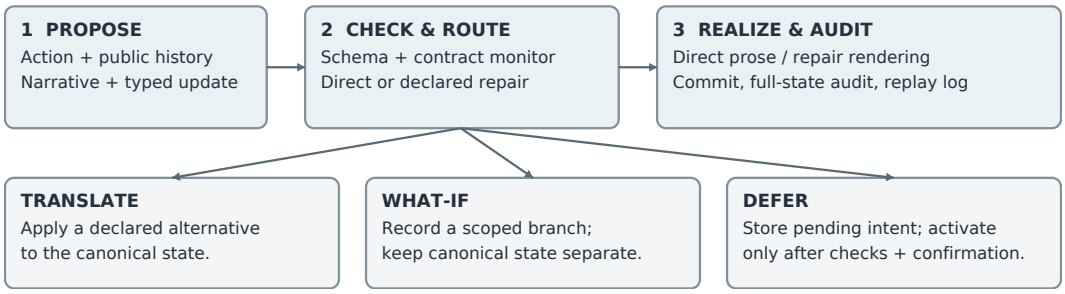


Figure 1: The implemented call and decision order. The initial narrator produces prose and a delta together. A second call renders a detected conflict's treatment; the state-level check does not independently certify either paragraph.

A contract $\mathcal{C}$ specifies typed facts, rules, and declared repair policies. The reference rule language has three forms: immutable fixes a field to a value; forbids disallows a field value; and requires conditions one field value on other fields. Each world has five rules, not five distinct rule-language primitives. Ordering is encoded through state prerequisites rather than a general temporal logic. Equality predicates are part of the frozen v0.2 semantics, including the evidence rule examined in Section 5.2.

An execution conflict occurs when $A(s_t \oplus \Delta_t, \mathcal{C}) \neq \varnothing$, where A is the full-state rule audit. If $A(s_t, \mathcal{C}) = \varnothing$, the candidate introduces the conflict; otherwise violations may be inherited. This is a property of a proposed update under an explicit contract, not a diagnosis of the model's knowledge. An infeasible request with an empty, valid update is not an execution conflict.

It is useful to distinguish two checks. The full-state audit $A(s, \mathcal{C})$ returns every rule violated by s. The implemented precommit monitor $M_\Delta(s, \Delta, \mathcal{C})$ inspects rules whose target fields occur in Δ. A correct admission test for a full-state contract would satisfy

$$M(s_t, \Delta_t, \mathcal{C}) = A(s_t \oplus \Delta_t, \mathcal{C}). \tag{1}$$

The v0.2 implementation does not satisfy this identity in general. Keeping precommit decisions and post-turn audits separately in the trace makes that defect observable rather than concealing it in an aggregate success label.

3.2 A declared repair policy

If the precommit monitor finds no conflict, StoryHARN applies the candidate delta directly. Otherwise it reads the repair policies attached to the violated rules. A policy can request an IF branch, deferral, or a published alternative delta. The selection priority is explicit: a branch request takes precedence; otherwise any deferral request defers the whole action; otherwise the policy combines declared alternative deltas and rechecks the result. An absent or invalid translation falls back to deferral. This is a finite priority policy, not a search for a semantically optimal rewrite. It cannot establish that the chosen repair is closest to the player's latent intent.

A translation applies only the declared alternative fields, potentially discarding other parts of a request. A protected bell's destruction might become a warning that increases narrative pressure. The trace label intent_preserved_as records the policy's interpretation, not a human judgment that it succeeded.

An IF branch records its parent turn, local state, and allowed rule exceptions. Creating it leaves the canonical state unchanged. The v0.2 runner records whether its initial state violates non-exempt rules; the continuation API validates a proposed next branch state against those rules before mutating it. Thus branch creation, branch validity, and successful continuation are distinct events. We do not assume that recording a branch makes it a valid alternative world.

Table 1: State-level meanings of the three repair families. These are implemented interfaces; their empirical coverage differs in the observed run.

Outcome	Recorded operation	What remains to be checked
Translated	Apply a rule-declared alternative delta instead of the original candidate.	Full-state validity and whether its visible consequence preserves the desired action.
IF branch	Copy state into a parented branch and record specific rule exceptions; retain the mainline.	Non-exempt rules, isolation, and subsequent branch continuation.
Deferred	Retain the mainline and store the original delta, blockers, and confirmation requirement.	Eligibility after later changes, fresh validation, and explicit confirmation.

A deferred record retains the original delta, blockers, and a confirmation requirement. A successful later recheck makes it eligible. A separate confirmation API revalidates current state before committing. In v0.2, calling this API counts as confirmation by default; the caller need not supply an explicit confirmation flag. Recording an intention does not show that its prerequisite will ever be achievable, and the fixed episodes never call this API.

3.3 The boundary of a safety claim

Suppose the initial state satisfies the contract, every mainline update is admitted only after a sound full-state check, and branch creation and deferral leave the mainline unchanged. Then every resulting mainline state satisfies the contract: direct and translated updates pass the check, and the other outcomes preserve the previous valid state. This elementary induction is a property of the specified interface under its assumptions, not a correctness proof of the v0.2 implementation. The monitor gap in Section 5.2 violates the full-state-check assumption. No corresponding guarantee is made for free-form prose, interpretation accuracy, or the eventual usefulness of a deferred action.

3.4 Traces and information barriers

Traces retain candidates, rule hits, dispositions, applied updates, before/after states, branch or intention identifiers, final responses, and post-turn audits. Call metadata records attempts, usage, latency, requested model, and supported parameters. A manifest hashes source files and frozen data. Expected action effects are available only after the decision, never to candidate generation or repair selection.

These traces make the structured decisions replayable. They do not expose hidden reasoning or guarantee that a closed provider will generate the same paragraph on another date. Fixture traces used for deterministic tests are marked separately and rejected by the paper-facing real-model analysis. The experimental materials contain authored worlds rather than production user conversations, following the emphasis on dataset scope and documentation in Gebru et al. (2021); Mitchell et al. (2019).

4 Benchmark and Experimental Protocol

4.1 A small, frozen stress test

StoryHARN-Bench v0.2 contains 12 original, researcher-authored worlds with AI assistance. Each has 20 fixed Chinese-language actions: 15 progression actions interleaved with five pressure actions. The pressure actions test early access, role replacement, premature resolution, damage to a protected object, and an ending without acknowledged consequences. The worlds vary their premises, characters, places, and wording, but share a state schema and action grammar. They are controlled surface realizations of related challenges, not an independent sample of genres or real user behavior.

The release has 240 authored actions. Each action includes an evaluator-only expected delta. The five rules per world check role continuity, access permission, evidence before resolution, a protected state, and consequences before ending. The release manifest records ordered worlds and hashes of the actions and rules. These materials were frozen before this comparison; this was a local version freeze, not a public preregistration. After inspecting the experiment, the 12 worlds should be treated as a development set for subsequent revisions.

All configurations receive the same fixed actions. Since histories diverge, later actions may be redundant or premature in one condition. Fixed replay controls inputs but does not adapt to the resulting story; twenty turns do not establish arbitrary long-horizon consistency.

4.2 Five configurations

Ordinary generation receives the premise, current canonical state, state-field instructions, and the action. Long memory adds all previous public turns, including their responses and state digests; it is a history control, not a retrieval or memory-compression system. Rule prompt additionally includes the public contract as serialized text. We retain the code identifier graph_prompt for trace matching, but no graph algorithm is used. These three configurations commit every schema-valid candidate delta without runtime repair. Only rule prompting supplies the full public contract among these three controls; pooled violations cannot all be attributed to failure despite supplied rules.

Hard rejection uses the same categories of context as rule prompting and leaves the state unchanged when the incremental monitor detects a conflict. StoryHARN uses that monitor with the three repair families. For hard rejection and StoryHARN, a separate narrator call describes each detected rejection or repair. All conditions use the same requested model and action set.

Final candidate instructions differ: rule prompting requests constraint adherence, hard rejection requests an explanation of conflicts, and StoryHARN says the runtime will handle conflicts. This is an end-to-end configuration comparison, not a runtime-only causal ablation. Prompt-induced caution can affect candidates before monitoring.

4.3 Model access and execution

The run used Codex CLI version 0.154.0-alpha.6.2, requested model gpt-5.6-sol, and low reasoning effort. Each completion used a fresh, ephemeral wrapper session. Tool use, external search, project instructions, and persistent memories were disabled; the recorded invocations contain zero tool events. The CLI did not expose the returned model identity, so these are wrapper-level observations rather than results for an independently verified bare-model endpoint.

Temperature and generation seed were not sent because this backend did not support them; an output-token cap was not enforced. The root seed controls experiment bookkeeping and statistical resampling, not model sampling. We used one rollout per world/configuration. Up to three episodes ran concurrently, with sequential turns within each episode. Consequently, reported latency reflects this environment and provider behavior, not a controlled hardware comparison.

The failure policy allowed one retry for candidate or repair calls. Failed attempts remain recorded; terminal failures would retain explicit failure and subsequent no-call entries instead of silently dropping an episode. All 60 episodes completed. The run made 1,253 CLI invocations: 1,203 candidate attempts and 50 repair/rejection-surface calls. Three failed provider attempts recovered under the retry policy. No terminal failure, missing cell, duplicate cell, or unexpected cell remains in the 1,200-turn grid. Dollar cost is unavailable and is not reported as zero.

4.4 Measurements and uncertainty

State safety is the fraction of turn records whose committed canonical state has no post-turn rule violations. Survival@20 is the fraction of worlds with no such violation over the

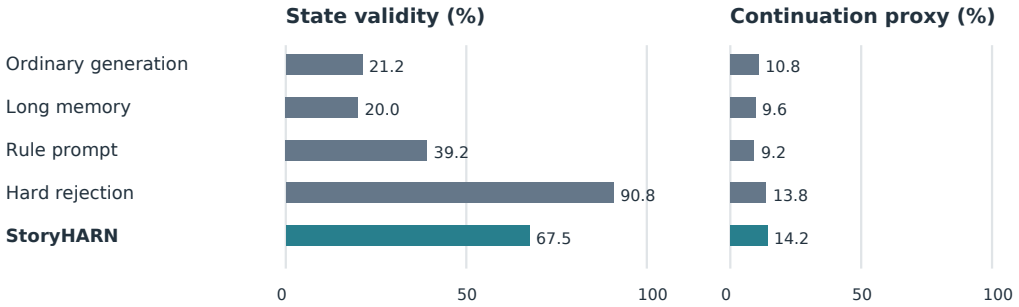


Figure 2: Complete v0.2 experiment: 240 turns per configuration, across 12 worlds. Both axes run from 0 to 100%. The continuation proxy combines candidate-effect matching and a trace-level audit; it is not a human assessment. Safe episodes (top to bottom): 0, 0, 0, 10, 5 of 12; runtime rejections: 0, 0, 0, 30, 0. Paired-world uncertainty appears in Table 2.

entire episode. A persistent illegal state is counted at each subsequent turn where it remains illegal; state safety is not a count of distinct narrative mistakes.

Candidate-effect coverage compares the proposed delta with the authored expected fields and values. It is evaluated after the runtime decision and does not assess the repaired prose. A separate continuation audit checks the recorded disposition: valid direct updates, recorded alternatives, scoped branches, or intentions with replayable activation conditions. We report the conjunction of this audit with strictly positive candidate-effect coverage as an action-continuation proxy; its code name is agency_preservation_rate. The name does not turn it into a measurement of human agency. A matched absolute value can reward an expected state transition, while a legitimate earlier discovery can receive no credit.

We also report runtime rejection, repair-family counts, wrapper-reported tokens, and latency. For each comparison, the unit of paired resampling is the world: 10,000 bootstrap draws of 12 matched worlds give descriptive 95% intervals. We do not treat the 1,200 correlated turns as independent observations. Intervals are unadjusted summaries for this fixed release, not population-wide claims or stability estimates over repeated model generations.

5 Results and Failure Analysis

5.1 Hard rejection is the strongest safety baseline

Figure 2 reports all five configurations. StoryHARN passes 162 of 240 structured-state checks, versus 94 for rule prompting and 218 for hard rejection. Five StoryHARN episodes remain safe for all twenty turns, compared with ten under hard rejection. Adding history alone does not improve state safety in this run. The authored actions, wrapper, and equality rule delimit these observations.

Table 2 retains the strongest comparison. The safety difference between StoryHARN and hard rejection is -23.33 percentage points, with a paired-world interval of $[-41.67, -4.58]$. Their action-continuation proxy differs by only $+0.42$ points, with an interval spanning zero. The data do not show that the repair system trades a clear improvement in continuation for its lower state safety. Nor does the positive proxy difference from rule prompting establish a human preference: it depends on exact authored field matches and the runtime's own continuation definition.

5.2 A dependency change bypasses the incremental monitor

All 78 unsafe StoryHARN turns and all 22 unsafe hard-rejection turns violate only evidence_before_resolution. The rule requires evidence_count == 3 whenever ending_unlocked is true. However, the incremental monitor checks this prerequisite only when

Table 2: StoryHARN minus each comparator, in percentage points, with 95% paired-world bootstrap intervals. The proxy is agency_preservation_rate; the separately computed joint safety/proxy metric has the same point values but is not substituted here.

Comparator	State safety difference [CI]	Proxy difference [CI]
Ordinary generation	+46.25 [31.67, 60.83]	+3.33 [−1.25, 8.75]
Long memory	+47.50 [33.33, 61.67]	+4.58 [0.00, 10.42]
Rule prompt	+28.33 [11.67, 45.42]	+5.00 [0.83, 10.00]
Hard rejection	−23.33 [−41.67, −4.58]	+0.42 [−5.00, 6.25]

Observed case: moon_archive, StoryHARN, turn 10.
Before: ending_unlocked = true; evidence_count = 3.
Candidate delta: {evidence_count: 4}.
Frozen rule: unlocked ending ⇒ evidence count = 3.
Incremental check: the target ending_unlocked is absent from the delta; its prerequisite is not checked.
Decision: direct commit.
Full-state audit: the ending remains unlocked with four pieces of evidence; the equality rule fails.

Figure 3: A real trace distinguishes an admission decision from the validity of its resulting state. Fixing dependency coverage and changing equality to a threshold are different interventions.

ending_unlocked appears in the proposed delta. Once an ending is unlocked, a delta that only increases evidence can bypass the check.

Figure 3 shows a concrete instance from moon_archive, StoryHARN, turn 10. The state already has an unlocked ending and three pieces of evidence. The narrator proposes a fourth piece. The precommit monitor returns no violation, the runtime records a direct commit, and the post-turn audit flags the resulting state. The apparent disagreement is fully explained by the two checks inspecting different sets of rules. Later turns can inherit the same error, inflating turn-level violations without introducing a different rule failure each time.

Two issues must remain separate. The monitor should evaluate every applicable rule over the proposed full state, including rules whose dependencies change. Independently, a designer might intend "at least three pieces of evidence" rather than "exactly three." The latter changes the benchmark's semantics; it is not a harmless correction to the reported score. We preserve the original rule and observations. This paper does not substitute a corrected replay for a new end-to-end model run: after state decisions change, the archived model outputs are no longer responses to the new histories.

Hard rejection shares this checker and is also imperfect. Its higher score reflects its observed candidates and trajectory, not a general solution to consistency. Conversely, StoryHARN's lower score does not imply incoherent paragraphs; no independent prose-to-state judgment was made.

To separate missed checks from inherited errors, we audited each archived candidate against its own original pre-state without changing the contracts or propagating hypothetical decisions. Among the 618 turns starting from a valid state, 105 candidates would violate a full-state rule. The incremental checker missed every violation in 14 of these candidates and additional violations in two more. Dependency-aware checking catches all violations in these 618 candidates while evaluating 611 rule instances, versus 3,090 for full checks. A matched runtime comparison and forced lifecycle tests isolate implemented capabilities without establishing player benefit (Appendix G). These are post-hoc diagnostics, not new policy rollouts.

Table 3: Separate v0.3 development follow-up. Every row has 240 planned turns. Safe/240 retains failures and no-calls; safe/committed is a conditional diagnostic, not the primary denominator. Proxy denotes the structured action-continuation proxy, not human agency.

Configuration	Committed	Safe/240	Safe/committed	Proxy/240
Ordinary generation	218	47 (19.58%)	47/218	25 (10.42%)
Long memory	208	46 (19.17%)	46/208	17 (7.08%)
Rule prompt	205	128 (53.33%)	128/205	23 (9.58%)
Hard rejection	195	195 (81.25%)	195/195	26 (10.83%)
StoryHARN	194	194 (80.83%)	194/194	29 (12.08%)

5.3 Implemented repairs are not equally exercised

Of StoryHARN's 240 turns, 220 are direct, ten translated, and ten deferred. No IF branch is created. All ten deferred intentions remain pending at the end; none reaches confirmed execution. Thus the experiment gives observations of translation and pending-action creation, but no estimate of IF success or deferred activation success. Branch validity has a zero denominator and is undefined, not 100%.

A translation in snow_crossing, turn 11, illustrates both its operation and its limits. The narrator proposes destroying the protected warning bell. The runtime substitutes a declared update that keeps the bell intact and changes narrative pressure to a warning to travelers. A second call renders that consequence. The repair applies the declared alternative, but the incoming canonical state already violates the evidence-equality rule; translation does not repair that unrelated historical error. A local alternative can therefore be correctly applied without making the full state valid.

Finally, runtime labels do not capture every visible refusal. In moon_archive, turn 1, the response says the player cannot enter without permission while the runtime labels the nonviolating candidate direct. Zero runtime rejections under StoryHARN therefore does not mean that no user-facing refusal occurred. Evaluating whether a repair feels responsive requires reading the output, not only counting dispositions.

5.4 Operational measurements

Mean recorded latency is about 20–21 seconds per turn across configurations (Table 4). These figures include wrapper overhead and calls under the observed concurrency. The additional repair calls do not yield a controlled latency estimate because different prompts, outputs, caching, and service conditions also contribute. Token counts are wrapper-reported input plus output tokens, not an estimate of dollar cost or only story text.

5.5 Revised-runtime development follow-up

We subsequently ran the same 12 development worlds under v0.3, with full-state checks, evidence thresholds, matched candidate instructions for rule prompting/hard rejection/StoryHARN, and additional safeguards (Appendix A.1). These are joint changes, not a single-factor ablation. Collection spans four bounded execution windows with the original CLI version. All 60 episode records are present, but only 1,020 of 1,200 logical turns committed: 14 terminal provider failures and 166 subsequent no-call entries remain in the denominator. There were 1,086 CLI invocations, including 41 failed attempts. Failure classification follows the frozen wrapper; it does not establish a network root cause.

Both hard rejection and StoryHARN have zero rule violations among their committed mainline states, and nine of twelve episodes survive safely through turn 20 in each condition. With failures retained, StoryHARN minus hard rejection is -0.42 percentage points in safe-turn rate (paired-world 95% interval $[-19.58, 21.67]$) and $+1.25$ points in the continuation proxy ($[-3.75, 6.67]$). These uncertain differences do not establish superiority or equivalence. The safe-turn difference against rule prompting is $+27.50$ points ($[3.33, 50.83]$), conditional on this small development set and observed failure pattern.

The 194 committed StoryHARN turns comprise 183 direct, six translated, two IF-branch, and three deferred decisions. Both new branch records satisfy their non-exempt checks, but there are zero branch-continuation steps; all three deferred intentions remain pending. Thus all three repair dispositions occur, while successful continuation remains untested. A separate offline audit replays all 1,020 committed transitions and reconstructs all 1,086 prompt hashes. This strengthens record-level verification without supplying evidence of narrative quality or user benefit.

Separate human evaluation of narrative consistency. Four readers each rated 50 items from a separate 100-trace, three-turn corpus, yielding 200 ratings and two readers per trace. The organizer confirms independent completion. With the primary supplied R1 table, readers agree on 97/100 semantic scores, 94/100 causal scores, and all actionability judgments; both deem 91 traces actionable. An alternative R1 source gives 94/100 and 97/100 semantic and causal matches. These exploratory results concern different materials and failures beyond execution validity; they do not test conflict repair or memory improvement. Provenance, score distributions, and source sensitivity are retained in Appendix F.

6 Limitations and Implications

The monitor defect and equality rule materially affect this small comparison. The worlds share a grammar, all actions and responses are Chinese, and only one wrapper configuration and one rollout per world were observed. Differing candidate instructions prevent runtime-only causal attribution.

Execution validity is relative to declared rules, not all narrative truth. We do not measure internal rule understanding or solve omitted-fact retrieval. State checks omit prose-only facts, while candidate-effect coverage can penalize sensible departures from the authored trajectory. The continuation audit evaluates records, not willingness to play. A branch can abandon the desired world, and an intention can stay pending forever. These limit the evaluation target itself.

No human ratings of the five-configuration repair experiment have been collected. A separate study of three-turn narratives is reported in Appendix F; it does not assess StoryHARN repair efficacy. The proposed matched-repair protocol in Appendix D remains unexecuted. We consequently make no claim about emotional engagement, enjoyment, perceived agency, retention, or commercial outcomes. An implemented product can motivate this problem, but its existence is not a controlled evaluation of the harness.

The diagnostic implication is concrete: admission and full-state audits should agree, and continuations should be tested across their lifecycle. Separately labeled engineering checks appear in Appendix A.1; the v0.3 model follow-up is reported in Section 5.5. Its corrected checker, changed contracts, matched prompts, and safeguards cannot be disentangled by cross-version score differences. Provider failures further limit comparisons. Reusing these worlds yields development results; generalization needs held-out situations.

7 Conclusion

StoryHARN addresses execution conflicts at the proposal-to-commit boundary: supplied rules constrain which state changes enter the mainline, while alternatives, branches and deferred intentions receive explicit records. The original 1,200-turn comparison reveals a dependency-checking defect and favors hard rejection. A separately versioned development follow-up removes observed committed-state violations in both monitored configurations and exercises all three repair dispositions, but does not establish a continuation-proxy advantage over rejection. These execution results establish neither improved recall nor better player-intent fidelity than rejection.

Ethics Statement

The benchmark uses original, researcher-authored synthetic worlds with AI assistance. It contains no production user conversations, payment records, copyrighted source fiction, or personal research-participant data. A separate four-reader study is reported on organizer confirmation of human origin and independent completion. Consent, recruitment, compensation, and applicable institutional review records were not supplied; these are not inferred from independence. Human assessment of the five-configuration repair experiment has not begun. Interactive stories may invite attachment or self-disclosure, but this work provides no evidence of mental-health benefit and should not be presented as such. Private product data and identifying deployment information are excluded from the anonymous research artifact.

Reproducibility Statement

Both versions retain 12-world × five-configuration × 20-turn trace grids with pinned code, rules, and actions. In v0.2, all 1,200 runtime transitions replay and all 1,253 recorded prompt hashes reconstruct. In v0.3, all 1,020 committed transitions replay and all 1,086 prompt hashes reconstruct; the remaining 180 grid entries are explicit failures or no-calls. Forced-input fixtures remain distinct from these model observations. Appendix E states versioning requirements. Separate-study rating transcriptions and frozen materials are included; they are not labels for either repair-experiment grid.

AI Use Disclosure

Generative-AI tools, including Codex, assisted with framing, literature discovery, evaluation design, code, synthetic materials, proof drafting, analysis, and manuscript preparation. Model-generated experimental outputs and their access limitations are described in Section 4. The organizer confirms independent human completion of R1–R4; Appendix F preserves provenance and source discrepancies. The authors retain responsibility for verifying all claims, code, and artifacts.

References

Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. Safe reinforcement learning via shielding. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 32, pp. 2669–2678, 2018. doi: 10.1609/aaai.v32i1.11797. URL https://ojs.aaai.org/index.php/AAAI/article/view/11797.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé, III, and Kate Crawford. Datasheets for datasets. Communications of the ACM, 64(12):86–92, 2021. doi: 10.1145/3458723. URL https://arxiv.org/abs/1803.09010.

Subbarao Kambhampati, Karthik Valmeekam, Lin Guan, Mudit Verma, Kaya Stechly, Siddhant Bhambri, Lucas Paul Saldyt, and Anil B Murthy. Position: LLMs can't plan, but can help planning in LLM-modulo frameworks. In Proceedings of the 41st International Conference on Machine Learning, volume 235 of Proceedings of Machine Learning Research, pp. 22895–22907, 2024. URL https://proceedings.mlr.press/v235/kambhampati24a.html.

Klaus Krippendorff. Content Analysis: An Introduction to Its Methodology. SAGE Publications, 4 edition, 2018. URL https://us2.sagepub.com/en-us/nam/content-analysis/book258450.

Yingpeng Ma, Jianhao Yan, Bei Shi, Ka Hou Kam, Runnan Wang, Xuebo Liu, Yulong Chen, Yue Zhang, and Derek F. Wong. Can LLM agents stick to the script? modeling commitment in interactive narratives. In Forty-third International Conference on Machine Learning, 2026. URL https://openreview.net/forum?id=JoJUWsQBp0.

Michael Mateas and Andrew Stern. Façade: An experiment in building a fully-realized interactive drama. In Game Developers Conference, 2003. URL https://users.soe.ucsc.edu/~michaelm/publications/mateas-gdc2003.pdf.

Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. In Proceedings of the Conference on Fairness, Accountability, and Transparency, pp. 220–229, 2019. doi: 10.1145/3287560.3287596. URL https://doi.org/10.1145/3287560.3287596.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. In Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology, 2023. URL https://arxiv.org/abs/2304.03442.

Mark O. Riedl and Vadim Bulitko. Interactive narrative: An intelligent systems approach. AI Magazine, 34(1):67–77, 2013. doi: 10.1609/aimag.v34i1.2449. URL https://ojs.aaai.org/aimagazine/index.php/aimagazine/article/view/2449/2353.

Mark O. Riedl, C. J. Saretto, and R. Michael Young. Managing interaction between users and agents in a multi-agent storytelling environment. In Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 741–748, 2003. doi: 10.1145/860575.860694. URL https://sites.cc.gatech.edu/fac/riedl/pubs/riedl-young-aamas03.pdf.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In Advances in Neural Information Processing Systems, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

Takuma Yoneda, Jiading Fang, Peng Li, Huanyu Zhang, Tianchong Jiang, Shengjie Lin, Ben Picker, David Yunis, Hongyuan Mei, and Matthew R. Walter. Statler: State-maintaining language models for embodied reasoning. In IEEE International Conference on Robotics and Automation, 2024. doi: 10.1109/ICRA57147.2024.10610634. URL https://arxiv.org/abs/2306.17840.

A Reference Policy and Continuation Lifecycle

The policy takes the current canonical state, candidate delta, detected rule violations, and the public contract. It does not take an authored expected action effect. For every violated rule, it reads the declared repair kind; if no declaration exists, an immutable rule defaults to IF branching, and other unsupported conflicts default to deferral. The finite selection procedure is:

1. If no violation is detected, return direct and the candidate delta.
2. If any violated rule requests IF branching, snapshot a branch, naming exactly the rules whose IF policies permit exceptions. Keep the canonical state unchanged and record any non-exempt branch violation.
3. Otherwise, if any violated rule requests deferral, store the complete original delta with its blockers and a confirmation requirement.
4. Otherwise, combine the rule-declared translation deltas in deterministic order. If any is missing or the resulting update fails the monitor, defer. If it passes, apply only the alternative delta.

This priority specifies behavior when a candidate conflicts with several rules. It does not prove that the priority is optimal, or that all combined alternatives are semantically compatible. Conflicting assignments in authored alternative deltas require particular care when extending the contract language.

The continuation API performs separate operations. To continue a branch, it validates the candidate delta against the closed state schema, checks the resulting branch state against all non-exempt rules, and mutates only that branch. Unknown exception identifiers are rejected. To confirm a deferred action, v0.2 requires an eligible record, schema-valid data, and fresh precommit and full-state checks, but treats a separate API invocation as confirmation by default. Its optional confirmation flag uses a truthiness check rather than requiring a literal boolean. A failed recheck returns the action to pending instead of committing an outdated permission. These API definitions describe implementation capabilities; no naturally generated IF continuation or deferred confirmation appears in the reported model run.

A complete mechanism test should cover success and failure paths separately: branch creation, two subsequent branch updates, mainline isolation, forbidden exception widening, prerequisites becoming available, explicit confirmation, loss of a prerequisite before confirmation, and repeated confirmation. Passing such deterministic tests is evidence about those interfaces, not a substitute for sampled model behavior or player judgments.

A.1 Revised implementation: separate engineering evidence

The separately frozen v0.3 reference checks the complete merged state and rejects an invalid controlled-runtime initial state. It publishes changes atomically. Branch creation and continuation must preserve all non-exempt rules and cannot widen the scope fixed at creation. Conflicting translation assignments fall back to deferral. Deferred eligibility is rechecked when conditions change; confirmation requires a literal true flag, a fresh validation, and a record that has not already been consumed.

All 97 automated tests pass with no skips. Eight authored mechanism cases additionally produce 25 deterministic operations covering valid and invalid translation, two-step branch isolation, mixed conflicts, deferred activation, stale confirmation, the original equality-rule regression, and the new threshold boundary. Their manifest labels them controlled_non_empirical_mechanism_test: they make zero model calls and collect zero human ratings. These cases test known interface paths; they are not estimates of natural triggering or a proof of arbitrary program correctness.

The new development contract uses an explicit evidence threshold of at least three, while regression tests retain the original equality rule. The 240 authored action definitions are unchanged. Candidate instructions are matched across rule prompting, hard rejection, and StoryHARN in v0.3, but resulting histories can still differ. These simultaneous revisions

preclude attributing a future cross-version score difference to the checker alone. We distinguish natural repair occurrence, forced mechanism coverage, and subsequent committed continuation throughout.

A.2 Unchanged-contract single-step diagnostic

Each of the 1,200 original records supplies its original state and model-proposed delta. An independent full-state implementation agrees with applying the original rule checker to the entire merged candidate state. The original equality contracts are retained; no counterfactual state is carried to the next record. There are 669 invalid candidate states, versus 627 invalid actually committed states, because rejection and repair can make those quantities differ.

Across all records, the incremental checker completely misses violations in 498 candidates and partially misses them in 80 more. These counts are dominated by inherited errors: of 1,032 missed rule occurrences, 1,000 already existed in the original pre-state and 32 are newly introduced. All 32 new misses concern prerequisites: 28 evidence rules and four access rules. Restricting attention to the 618 valid original pre-states yields the 14 complete and two partial misses reported in the main text. Twelve diagnostic tests verify the rule interpretation, non-mutation, denominators, and byte-identical record reproduction. These are post-hoc checks on observed candidates, not new model generations.

B Frozen Data and Evaluation Details

The twelve world identifiers are cat_commons, snow_crossing, harbor_mystery, moon_archive, last_train, underwater_court, memory_theater, cloud_orchard, old_arcade, sunless_city, tea_house, and starship_garden. Their premises are original synthetic settings. Shared action positions help match pressure across configurations but narrow the benchmark's diversity.

Let q_{wt} denote the positive candidate-effect coverage indicator and c_{wt} the replayed continuation indicator for world w, turn t. The action-continuation proxy is the mean of $q_{wt}c_{wt}$. Its contract-safe variant additionally requires zero post-turn violations. Both have the same point values in this run. The analysis samples their intervals separately, so their endpoints can differ slightly: against rule prompting, the proxy difference interval is $[0.83, 10.00]$ points while the contract-safe variant gives $[0.42, 10.00]$. We retain the named proxy throughout the main table instead of selecting a favorable interval.

The expected-effect measure compares absolute field values in a proposed delta with the authored oracle. It is not a comparison of complete resulting states. In particular, not changing a field already satisfied earlier can yield no credit for that expected update. This distinguishes the metric from successful completion of a natural-language goal. Audit indicators also depend on the correctness and coverage of the contract; they cannot establish the truth of an unrepresented fact.

Each paired bootstrap draw samples the same twelve world indices with replacement for both configurations and averages their per-world metric differences. With only twelve related worlds and one rollout per cell, these intervals quantify within-release sensitivity. No adjustment for multiple comparator/metric combinations is applied, and no population-level significance claim is made. Provider failures would remain in the denominator through failed or no-call records. The actual run has no terminal failed cells, while the three recovered call failures remain part of the execution accounting.

C Execution Accounting

Table 4 lists the observed calls and wrapper-reported resource use for the preserved run. These measurements include recovered attempts and do not imply a monetary cost estimate.

Table 4: Observed execution accounting. Invocations include retained failed attempts and successful recovery. "Surface" calls render detected repairs or rejections. Latency and token means use the existing frozen analysis.

Configuration	Invocations	Surface calls	Mean latency (s)	Mean tokens
Ordinary generation	241	0	20.46	5,127
Long memory	241	0	20.15	6,786
Rule prompt	241	0	20.42	6,995
Hard rejection	270	30	21.47	7,555
StoryHARN	260	20	20.40	7,392

D Prepared Human Assessment Protocol

Status: zero ratings for this matched-repair pilot. The separate R1–R4 study below uses different materials and does not complete this protocol. The current prospective pilot compares StoryHARN with hard rejection on 60 prespecified world–action pairs: five stress actions in each of 12 worlds. Three independent adult Chinese readers would each complete three sessions of 20 pairs, yielding 180 pair assessments, not 180 participants. Earlier four-control packets are archived preparation artifacts, not the current collection protocol or completed evidence. As of the material audit, 46 of the 60 pairs have both required committed outputs from the v0.3 run. The remaining pairs are unavailable because of retained failures and no-call entries. These 46 cannot replace the prespecified sample. Recruitment, institutional determination, and a validated collection pipeline remain pending.

Readers would view each candidate's action, relevant state, preceding context, and randomized A/B response. Rollout histories can differ across methods; sharing a world and action does not imply identical preceding states. Separate A/B/tie judgments assess prose–state consistency (primary), alignment with the stated action, and whether the response provides an actionable continuation (secondary). Observers did not choose the action, so they must not be asked whether "my intention was understood." That question belongs to a separate active-play study. Product-experience questionnaires and proposed recruitment counts are not harness comparison results. Method labels and mappings should be hidden from readers; a coordinator should retain private randomization keys. Unless analyst blinding is actually implemented, the study should report rater blinding, not double blinding.

The current plan requires three valid independent readers per pair and dimension, yielding 540 dimensional judgments if complete. Each of nine sessions includes three attention checks (27 in total); all three must be correct for that session's ratings to qualify. Missing or unrateable responses remain missing, not ties. Coverage and exclusions must be checked before unblinding. If the fixed sample cannot be completed, any exploratory subset requires an explicit prospective amendment and missingness report; model preferences cannot substitute for independent readers.

A completed pilot should report raw wins/losses/ties, coverage, exclusions, and nominal Krippendorff's alpha separately per dimension (Krippendorff, 2018). The planned estimate averages readers within pair, then five pairs within world, then the 12 worlds equally, coding ties as one half. A 10,000-resample world bootstrap with seed 20260919 supplies percentile intervals conditional on this small reader set; it does not establish population-level preferences. An offline statistics module has synthetic tests, but this does not validate participant provenance, collection, attention-check processing, or a completed human study.

E Reproduction and Revision Boundary

The preserved experiment is benchmark version 0.2.0-frozen and trace schema storyharn.harness_real_trace.v2. Its merged trace SHA-256 is

47b8c2a47f87824d8cdfa8a072ff2e065f9b06889ac0811c3fd4ae95ff04bd87.

The manifest includes hashes for source files and the closed 14-field state schema. Reproduction should first check these hashes and the complete 1,200-cell grid, then regenerate analysis from the recorded JSONL data. The recorded UTC run window is 18 September 2026, from 17:33:42 to 19:52:44. A new live run must use a new output directory and record its own backend, parameter support, attempts, usage, and timestamps.

Two forms of reproducibility must not be conflated. The deterministic runtime and analysis can be inspected and replayed on archived candidates. A new closed-provider generation may differ despite the same requested model name and prompts. In addition, replaying an old candidate after changing a previous state produces a diagnostic counterfactual, not the model's response to that revised history.

The monitor correction suggested by Equation 1 and a change from an equality predicate to an evidence threshold require separate change records. Testing the monitor while retaining the equality rule determines whether it catches dependency-only updates. Testing a threshold then changes which states the contract considers valid. Neither operation can overwrite the v0.2 score. Once the code and rules are revised, a new complete configuration grid is needed for an end-to-end comparison. Mechanism stress tests that force IF or deferred activation should have separate manifests and denominators from natural model-triggered events.

The anonymous research artifact should contain only the original benchmark, research prompts, runtime, analysis, manifests, traces, and documentation needed for verification. Credentials, account identifiers, private production prompts, real user logs, commercial materials, and blind-assessment answer keys do not belong in the public package. NCP-compatible specification loading is only an interface check: an external NCP evaluation would require its untouched evaluator and a distinct protocol. IF branches, which have no corresponding mainline success semantics in that protocol, must be evaluated separately.

F Separate Human Evaluation of Three-Turn Narratives

Design and scope. This study uses 100 synthetic three-turn narrative traces from a separate consequence-evaluation corpus (sample IDs H001–H100), not the 12-world, 20-turn StoryHARN experiments. One trace per root was selected from 100 roots spanning ten mechanism families, balanced across four generation interfaces and two model backends. These interfaces are separate from the StoryHARN repair comparison. Each trace presents initial facts and rules, an action, observation, and recall. The frozen forms hide generator identities and expected outcomes. Four readers, R1–R4, each assess 50 items; every item is assigned to two distinct readers. The resulting 200 item assessments are not 200 participants. The organizer confirms that all four readers completed the assessments independently, without using other reviewers' results or AI-prefilled scores. The blind form design and reported independent completion are distinct from an independently audited collection process.

Readers rate semantic consistency and causal responsiveness on 0/1/2 scales. Semantic 2 denotes consistency, 1 ambiguity, and 0 a clear contradiction; causal 2 denotes appropriate action handling, 1 partial response, and 0 an ignored or reversed response. They also judge whether another action can proceed without first correcting the story, and report confidence. Native exports, per-turn fact labels, and adjudication records were not supplied. We analyze the transcribed score tables, preserving missing information. This is a separate exploratory human evaluation, not a gold-label calibration of the automatic checker or evidence that StoryHARN outperforms hard rejection.

Provenance. The supplied summaries contain AI-origin wording, including AI-ASSISTANT; the organizer states that this wording is erroneous and confirms human authorship and independent completion. This correction is recorded separately rather than silently rewriting the original source history. Some supplied rationales share wording; the confirmed independence concerns the assessments, while the report-writing workflow has not been separately audited. No recruitment, consent, compensation, or institutional-review details are inferred or fabricated.

Table 5: Separate human study: reported distributions, using the inline R1 source. Counts for the two score dimensions are in 2/1/0 order. Each row represents one reader's 50 assessments.

Reader	Semantic 2/1/0	Causal 2/1/0	Actionable
R1	37 / 2 / 11	44 / 2 / 4	44
R2	40 / 5 / 5	46 / 2 / 2	47
R3	33 / 8 / 9	42 / 6 / 2	46
R4	40 / 4 / 6	44 / 2 / 4	45
Total	150 / 19 / 31	176 / 12 / 12	182

Source sensitivity and agreement. Two supplied R1 tables cover the same 50 IDs but differ in 11 rows across scores or confidence. Their version relationship remains unresolved. The inline table is the primary transcription; the attachment is an alternative-source analysis, not a fifth reader. With inline R1, exact semantic agreement is 97/100 and causal agreement 94/100; with the attachment these are 94/100 and 97/100. Actionability matches on all 100 items in either version: both readers say yes for 91 and no for nine. Thus 182 affirmative assessment records correspond to 91 distinct traces, not 182 independent successes. These are descriptive raw agreements; agreement does not establish correctness, and no adjudicated reference or chance-corrected reliability claim is made.

Qualitative implications. The supplied reports flag action recollections that conflict with earlier prose. H072 remains silent in turn one but later recalls an unsuccessful attempt to speak; H074 makes no renewed alliance proposal but recalls a rejected proposal; H028 declines repair but recalls a failed attempt. In each case both assigned readers give semantic 0 under either R1 source version. The two histories can have the same resulting state, motivating a distinction between state validity and faithful recall. These selected examples are not an estimate of error prevalence or evidence that StoryHARN repairs the problem. English descriptions paraphrase the exact Chinese materials retained in the supplement.

Reproducibility and inference boundary. The supplement's human_study/ directory contains the 200 transcribed assessments, alternative R1 table, allocation, all 100 frozen presentation packets, source/presentation hashes, organizer-confirmation record, and a standard-library analysis script. Scores are separated from the rule-defined oracles of both StoryHARN experiments. We do not pool these readers with the prospective matched-repair pilot, derive a StoryHARN-vs-rejection preference rate, or treat confidence as correctness. Establishing player-intent fidelity or repair benefit still requires assessments of the actual repair outputs under an appropriate protocol.

G Controlled Monitor and Repair Diagnostics

Evidence status and common inputs. These analyses were designed after inspecting the original experiments. They add no model generations, repair prose, human ratings, or held-out worlds. The archived 1,200 v0.2 candidates, their original pre-states, and exact-equality contracts remain unchanged. Runtime interventions use the separately archived v0.3 implementation but retain v0.2 rules. Each candidate is evaluated from its original pre-state; no counterfactual trajectory is presented as a newly generated story. Input and implementation hashes, executable scripts, per-row decisions, exclusions, and checks are in improvements/ in the supplement.

G.1 Dependency-aware checking under an unchanged contract

Let $D(r)$ be the set of fields read by a pure rule predicate r, including its target and prerequisite fields. For update Δ, let $R_\Delta = \{r : D(r) \cap \text{keys}(\Delta) \neq \varnothing\}$. If the pre-state is valid, evaluating R_Δ on the merged post-state is sufficient. More generally, with an exact

cache V_t of prior violations,

$$V_{t+1} = (V_t \setminus R_\Delta) \cup \{r \in R_\Delta : r(s_t \oplus \Delta) = \text{false}\}. \quad (2)$$

This follows because an unaffected predicate reads identical values before and after the update; affected predicates are re-evaluated. The identity requires complete dependency sets, fixed rules, exact overwrite semantics and correct cache initialization. Changing a rule requires invalidating the corresponding dependency information and cache. This is a correctness argument for ordinary dependency tracking, not a claim of a new learning algorithm or a guarantee about narrative truth.

Among 618 originally valid pre-states, 105 candidate post-states violate the original contract. Target-only monitoring completely misses 14 and partly misses two. Dependency-aware checking matches the full audit on all 618 rows, with 611 rule evaluations instead of 3,090. Across all 1,200 rows, dependency-only checking still misses inherited violations; retaining an exact prior-violation cache restores full agreement, with 1,046 affected-rule evaluations versus 6,000 full evaluations. Here the cache is initialized by a full audit of each archived pre-state; initialization, dependency selection and bookkeeping are excluded from these counts. The implementation scans rules to select dependencies. These counts therefore demonstrate reduced predicate evaluation, not measured runtime acceleration. An independent finite-domain oracle checks 1,728 state/update combinations, including target changes, prerequisite changes and inherited invalidity.

G.2 Matched single-step runtime interventions

We apply hard rejection, translation-only and full StoryHARN to the same 618 valid-pre-state candidates using one full-state checker and the original contracts. The other 582 rows are explicitly excluded because the corrected runtime refuses invalid starting states. A fresh context is initialized for each row; prior pending intentions and branches are not replayed. Translation-only uses the same declared alternatives, but rejects when any conflict lacks an alternative or their combined replacement is invalid. It neither creates branches nor stores pending intentions.

Table 6: Post-hoc interventions on the same 105 conflicting candidates. Counts describe runtime dispositions, not successful continued play. All three arms preserve canonical-state validity. The 513 non-conflicting candidates produce identical direct states across arms.

Runtime	Translated	Branch	Deferred	Rejected
Hard rejection	0	0	0	105
Translation-only	13	0	0	92
StoryHARN	13	24	68	0

Both alternative-enabled arms make the same 13 translations to exactly the same canonical states. Thus the additional capability measured here is branch and pending-intention creation, not better immediate translation. Zero canonical violations are expected from a shared full checker and valid pre-states; they do not distinguish the methods. The candidate pool combines trajectories generated under five original conditions and cannot estimate the rollout distribution of any new policy. We do not attach an independent-turn significance claim to these counts.

G.3 Lifecycle comparison with equivalent retry opportunities

Six forced synthetic scenario families are instantiated on the 12 existing worlds: access followed by retry/confirmation, withheld confirmation/retry, revoked access, duplicate retry/confirmation, hypothetical-branch continuation with an attempted scope violation, and translation followed by progress. The resulting 72 world/case combinations contain 684 operations across the three arms. Inputs are authored deltas; this is neither a new model experiment nor 72 independently sampled tasks.

When access becomes available, the controls can explicitly retry their original request in the same interaction slot in which StoryHARN confirms its stored intention. All three arms

reach the destination in all 12 such cases. Withholding confirmation/retry prevents entry, and revoking access blocks it. Repeated confirmation is rejected; repeated control requests leave the final location unchanged in this particular scenario. These tests therefore show no completion advantage for deferral over an allowed retry. They do not establish idempotence for arbitrary actions.

StoryHARN completes one scoped branch continuation per world without changing canonical state; all 12 attempts to violate the non-excepted access rule are blocked. Controls retry the same hypothetical-role request and reject it, since they do not implement branches. This isolates a supported interface rather than demonstrating player preference. Translation tests preserve the protected object and permit subsequent clue progress. Full before/after records and canonical/scoped-contract checks accompany every operation. Establishing utility still requires independent assessment of actual repair realizations and end-to-end episodes with comparable player opportunities and budgets.

G.4 Fresh realization feasibility pilot

A separate pilot generates new prose for six archived conflict cases: the first two distinct world IDs per repair family after sorting valid-pre-state candidates by world, step and source condition. Selection is post-hoc relative to the original experiment but fixed before these new realizations; the six cases cover three existing worlds. Each case shares its original state, action, candidate update and equality contract across hard rejection, translation-only and full StoryHARN. Each arm receives one realization call, with no retries and a fixed randomized call order. There are no new candidate generations or end-to-end trajectories.

All 18 planned realization calls complete; 16 satisfy the descriptive 80–180 Chinese-character instruction, and both length deviations remain in the data. The wrapper reports 98,170 input and 2,549 output tokens. The requested model is gpt-5.6-sol through CLI 0.155.0-alpha.2.6, with low reasoning effort and a 45-second timeout per invocation. Returned model identity is unavailable for all calls; sampling seed, temperature and output-token ceilings are not enforced. One JSON-only environment probe is excluded from the 18 realizations. Tool events are rejected, and none occurs. The artifact pins the input, runner, prompts and responses and retains sanitized invocation metadata.

There are only 12 distinct realization prompts: in each case, two arms select an identical treatment and receive identical prompts. Differences between those duplicate-prompt realizations are sampling variation, not evidence of a policy advantage. Complete outputs and empty blind-review packets are prepared, but no new human ratings have been collected. This pilot establishes execution feasibility only; completion and length compliance do not measure semantic accuracy, player intent, preference or generalization. It does not complete the separately prepared 60-pair protocol in Appendix D.

G.5 Prospective own-history rollout feasibility comparison

We separately freeze three newly authored scenarios, their code, analysis and execution order before collection: compound access prerequisites, revoked permission before retry, and combined evidence/responsibility prerequisites for ending. This local freeze is not external preregistration. Each scenario has three scripted player actions and three runtime arms. Each completed turn has one fresh candidate call and one realization call, without retries: nine planned episodes, 27 turns and 54 calls. Every arm generates from its own evolving state and realized public history, under the same full-state checker, rules and per-turn call budget. Private terminal goals are withheld from prompts; rule declarations are visible to all arms.

The final player action explicitly confirms and repeats the earlier request for every arm. If the full runtime has a pending intention, it checks confirmation; controls can retry. The candidate call is still made on confirmation turns to equalize call opportunities, and its delta is marked unused. This is not an efficiency test of confirmation. Permission revocation is a predeclared public event; appropriate refusal is reported separately from positive task

completion. Terminal failures stop an episode and retain subsequent no-call slots. The requested model and isolated CLI configuration are the same as the realization pilot.

The run completes 9/9 planned episodes with 54/54 attempted calls, 54 completed calls and 0 call failures. Table 7 reports the frozen endpoints; all completed turns undergo independent rule auditing and exact prompt/history reconstruction.

Table 7: New three-scenario feasibility pilot. "Conflicts" counts candidates violating the shared contract, not all natural-language requests that are infeasible. Endpoint validity concerns recorded state, not prose fidelity.

Runtime	Complete	Goal reached	Proper refusal	Conflicts
Hard rejection	3/3	2/2	1/1	0
Translation-only	3/3	2/2	1/1	0
StoryHARN	3/3	2/2	1/1	0

The full-runtime arm records 0 non-direct dispositions across 9 completed turns. The narrator handles the constraints without triggering the external repair mechanisms; these observations therefore provide no test of repair benefit. The sample is three authored feasibility scenarios, not a representative benchmark, a cross-model comparison or evidence of player preference. There are no independent-turn significance claims or new human ratings. We preserve all planned positions and do not expand the sample based on observed outcomes. The complete calls, state transitions, frozen evaluator and reproduction scripts are supplied separately from the original 1,200-turn experiments.