

PRE-FILING CITE AUDIT — THE NINE PITFALLS WE CATCH

An automated forensic pass on every citation, every rule reference, and every holding paraphrase before the document leaves the door.

Author: Richard L. Sanders · Utah Bar #15728 **Role:** Sole inventor and implementer
Version: 1.0 — 2026-05-15 **Companion papers:** [docs/whitepapers/01-deterministic-closed-corpus-verification.md](#) (AEGIS text-containment) · [docs/whitepapers/08-aegis-prime-structural-verification.md](#) (AEGIS PRIME disposition + binding) · [docs/whitepapers/05-v4-advocacy-algorithm.md](#) (V4 truth-table gate) **Source:** [feedback_pre_filing_cite_audit_mandatory.md](#) (standing-order memory), the Hymas filing trio at [docs/client-matters/hymas/{08,09,10}*.md](#), the audit-trail memos at [docs/client-matters/hymas/{AUDITOR-RESULTS, VERIFICATION-RECORD, UNANIMITY-CERTIFIED}.md](#), all in the public repository at [<https://github.com/Benchslap/Benchslap>](https://github.com/Benchslap/Benchslap)

Abstract

A legal brief that ships with one fabricated citation is sanctioned. A brief that ships with one wrong pin-cite is impeachable. A brief that ships with one rule number off by a subdivision is mocked. None of these outcomes are recoverable after submission. The forensic pre-filing cite audit is the single gate between filing-ready and filing-disaster.

This paper describes the architecture of that gate as it has matured through three Hymas-case submission cycles plus the OPC supplemental complaint in early 2026. It documents the nine specific failure classes the pipeline catches — drawn from the actual pre-filing audits run on real BenchSlap-drafted documents in 2026 — and the corpus + rule-text infrastructure that lets the audit run in seconds against a 3.3-million-opinion local cache.

The thesis is operational: every cite, every pin-cite, every rule reference, every holding paraphrase must pass a forensic verification before the document is treated as filing-ready. The verification is not a model's self-confidence check; it is a deterministic corpus query corroborated against a public mirror. When the corpus and the mirror disagree, the disagreement is the signal — *investigate before claiming*. The 2026-05-14 audit pass on the Hymas table of authorities caught a Knowlton-v.-Knowlton fabrication, a Risher-v.-Emerson

neutral-cite error, an URAP 23 page-limit misframing, and a Wilkerson-v.-Stalder subject-error — every one a TIER 1 filing failure had the document gone out as drafted. This paper documents what the audit looks for and why each class of failure is structurally invisible to a model's own pre-flight self-check.

1 • The standing order

`feedback_pre_filing_cite_audit_mandatory.md` lives in the operator's auto-memory hot ring:

NEW STANDING ORDER 2026-05-14: every filing-ready legal document MUST go through a forensic cite-verification audit BEFORE submission. Audit must include: (a) every neutral citation pattern verified against corpus, (b) every rule citation verified against rule text, (c) two-source corroboration mandatory. 2026-05-14 session caught Knowlton fabrications + Risher cite error + URAP 23 page-limit error in pre-flight that would have been TIER 1 filing failures.

The rule was promulgated the day a draft Hymas table of authorities was caught with fabrications in pre-flight. It applies to every filing-ready document from that day forward.

2 • The taxonomy of nine pitfalls

These are the specific failure classes the audit catches, each documented with a real example from a real pre-filing pass.

2.1 • Pitfall 1 — Fabricated case (no such opinion exists)

Example: *Knowlton v. Knowlton*, 2023 UT App 16 and *Knowlton v. Knowlton*, 2024 UT App 25. Both cited in an early Hymas TOA draft as standing for the proposition that an appellate court may treat factual findings as a "mathematical impossibility" when the trial court's arithmetic does not match the trial record. Neither neutral citation exists in the corpus. Neither exists in the public Utah Courts opinion search. The cases were fabricated by an AI drafting pass that synthesized a plausible-sounding citation for a real proposition.

Detection: corpus query `SELECT id, case_name FROM opinion_library WHERE neutral_citation = '2023 UT App 16'` returns 0 rows. Cross-check against utcourts.gov opinion search returns 0 rows. The agreement of zero matches across two independent sources is decisive.

Remediation: the proposition is real and well-supported — the actual cases are *Knowles v. Knowles*, 2022 UT App 47 (mathematical-impossibility framing) and *Bakanowski v.*

Bakanowski, 2003 UT App 357 (treatment of unsupportable Jones findings). Both substituted in commit history.

2.2 · Pitfall 2 — Case exists but neutral citation is wrong

Example: *Risher v. Emerson*, cited as 2017 UT App 250 in multiple Hymas drafts. *Risher v. Emerson* exists. Its neutral citation is 2017 UT App 216, not 250. The drafting AI substituted a different paragraph number into the cite position.

Detection: corpus query on 2017 UT App 250 returns a *different* case entirely. The case-name field of the corpus row does not contain "Risher" or "Emerson." This is a single-source check that fails decisively on its own.

Remediation: rewrite every Risher reference to 2017 UT App 216. Four instances across the URAP 10 motion and TOA were corrected pre-filing on 2026-05-14.

2.3 · Pitfall 3 — Hallucinated case at a real citation

Example: *McFarland v. McFarland*, 2015 UT 33. The neutral citation 2015 UT 33 resolves to a real opinion — but the real opinion is *Sawyer v. Department of Workforce Services*, an unemployment-benefits case unrelated to family law. The drafting AI attached the McFarland party name to a citation slot that happens to be filled by a completely different case.

Detection: corpus row for 2015 UT 33 returns `case_name = 'Sawyer v. Department of Workforce Services'`. The mismatch between the cited case name and the actual case name at the cited slot is the signal.

Remediation: *McFarland v. McFarland* exists at 2021 UT App 58. Substituted in all drafts.

2.4 · Pitfall 4 — Real cite, wrong subject

Example: *Wilkerson v. Stalder*, 329 F.3d 431. Real case. The proposition cited was about Social Security preemption. The actual case is the Angola-3 prison-conditions matter — completely different subject. The cite was real but the proposition it was attached to was nonsense.

Detection: opinion text-containment check (AEGIS, paper 01) reads the actual *Wilkerson v. Stalder* opinion and finds no language about Social Security preemption. The subject-mismatch flag fires.

Remediation: removed from filing; the proposition is correctly supported by *Crawford v. Gould* on the actual federal-preemption point.

2.5 · Pitfall 5 — Real cite to a case that has been overruled or superseded

Example: This is the failure mode AEGIS PRIME (paper 08) catches. A cite that is text-containment-clean and disposition-correct but to a case that has been overruled by a later decision. The audit pipeline queries `opinion_library` for `superseded_by_opinion_id IS NOT NULL` and flags any cited authority whose row points to a later overruling decision. The AI's claim becomes "this case used to say X" — usually not what the drafter intended.

Detection: AEGIS PRIME structural extraction (commits in 2026-04) populated `superseded_by_opinion_id` for 129 known-overruled Utah opinions. Citation of any of these triggers the SUPERSEDED_CASE block.

Remediation: cite the overruling case instead, or expressly note that the cited case is overruled and frame the proposition accordingly.

2.6 · Pitfall 6 — Rule number off by a subdivision

Example: URAP 4(b) cited where the controlling rule is URAP 4(c). Both subdivisions exist in URAP 4; (b) governs cross-appeals while (c) governs the timing trigger that was actually being argued. The drafting AI conflated them.

Detection: rule-text comparison. The audit queries `rules_library` for the cited rule subdivision, extracts the rule text, and compares the drafter's paraphrase against the actual subdivision's language. A subdivision mismatch produces a structural fingerprint that is easy to flag — the words "cross-appeal" in URAP 4(b) versus "trigger" in 4(c) are mutually exclusive markers.

Remediation: corrected to `URAP 4(c)` in the URAP 10 motion. Cross-document consistency check: the TOA and motion text now match.

2.7 · Pitfall 7 — Rule reframed against its actual text

Example: URAP 23(b)(2) cited as imposing a 15-page limit on motions. URAP 23 has no page-limit subdivision. The 15-page limit lives in URAP 24 for petitions for rehearing; URAP 23 addresses something else entirely (motion practice without the page-cap framing). The drafting AI cross-pollinated two rule numbers.

Detection: rule-text comparison. The audit reads URAP 23 in full and searches for "page" or "15" or "limit." The rule contains none of these as a binding word. The "15-page limit" claim attached to URAP 23 fails the substring containment check.

Remediation: removed the "15-page limit" framing entirely. The actual page-limit rule is found and cited in the right place.

2.8 · Pitfall 8 — Attribution flip (dissent quoted as majority)

Example: A panel-vote-vs-quote mismatch that AEGIS PRIME (paper 08) catches. The drafter quotes language from an opinion and frames it as the holding, but the language appears in the dissent. The text-containment check (AEGIS, paper 01) passes — the quote

really is in the opinion. The structural check fails — the quote is attributed to the wrong author.

Detection: AEGIS PRIME's `panel_vote` extraction stores `{ majority: [judge names], dissent: [judge names], concurrence: [...] }` per opinion. The audit queries which panel-member authored the quoted section and compares against the drafter's attribution. A dissent quoted as majority produces `ATtribution_Mismatch`.

Remediation: either re-attribute the quote to the dissent (with the framing "as the dissent observed") or find a majority statement that says the same thing.

2.9 · Pitfall 9 — Cross-document inconsistency

Example: the URAP 10 motion cites a case as supporting Proposition A. The companion TOA cites the same case as supporting Proposition B. The two propositions are not contradictory but the divergence reads as careless to a court that expects internal consistency between the brief and its supporting authority table.

Detection: cross-document grep on case names and neutral citations across the entire filing trio. The audit produces a matrix: for each cite, which document(s) cite it, with what parenthetical, on what pin-cite. Any case that appears in two documents with materially divergent parentheticals is flagged.

Remediation: unify the parenthetical or split into two distinct cites of the same authority for the two propositions.

3 · The audit architecture

The audit runs as a parallel sub-agent invoked at the moment a document is declared filing-ready. Its inputs:

- The three filing documents (markdown source of truth, not the PDF render)
- The local corpus (3.3 million opinions + ~200 rule-set rows under the 2026-05-15 multistate-rule-harvester tag plus Utah's URPC/URCP/URAP/URE captured 2026-05-14)
- A public-mirror fallback (utcourts.gov, public.law) for two-source corroboration
- The session-memory list of nine prior pitfalls (this paper's §2) as a watchlist

Its outputs:

- TIER 1 fixes applied (cite fabrication / wrong neutral / wrong subject / wrong subdivision — patched directly in markdown)
- TIER 2 fixes applied (pin-cite drift / parenthetical mismatch / cross-document inconsistency)

- Clean cite count (with sample)
- Unverifiable flags (anything that didn't resolve to a corpus or mirror hit — defers to operator)
- Structural issues (caption errors, missing certificate of service, page-numbering)
- Placeholder list ([BRACKETED] , <<<>>> , TODO markers that need operator fill)
- Confidence level (HIGH / MODERATE / LOW that filing-ready)
- Remaining gates

Each audit produces a memo at `_VERIFICATION-RECORD-<timestamp>.md` and an updated `_AUDITOR-RESULTS.md`. The trail is preserved so a future court inquiry into how the drafter satisfied themselves of the citations' accuracy can answer with file paths and timestamps.

4 · Why a model's self-check is insufficient

A drafting model can be prompted to "verify your citations before submitting." Three reasons that prompt does not produce reliable output:

(a) The model does not know what it does not know. A fabricated cite is plausible-sounding precisely because the model generated it from the distribution of plausible Utah-Court-of-Appeals citations. The model's self-check produces the same plausibility score on the fake cite as on a real one. Without an external source, there is no signal to distinguish them.

(b) Self-check inherits the same hallucination distribution. The model that wrote "Knowlton v. Knowlton 2023 UT App 16" is the model that will self-check it. If the source of the hallucination is a training-data lacuna or a sub-distribution drift, the same model on the same prompt will reproduce the same lacuna.

(c) Confidence is uncorrelated with correctness on long-tail facts. GPT-class models are reasonably calibrated on common facts (HIGH confidence corresponds to ~HIGH accuracy on common facts). On long-tail facts — the specific neutral cite of a 2017 Utah Court of Appeals case — calibration breaks down. The model can be 95% confident in a fabricated cite.

External corpus + rule-text comparison sidesteps all three. The corpus either has the row or it doesn't; the answer is binary and inspectable. The audit's job is to ask the corpus, not the model.

5 · The five-tier verification pipeline as background

The pre-filing audit sits on top of BenchSlap's existing five-tier verification pipeline (T0–T4 documented in `docs/whitepapers/01-deterministic-closed-corpus-verification.md`):

Tier	Source	Role in pre-filing audit
T0	Local cache (<code>verification_cache</code> , <code>opinion_library</code>)	Primary corpus check
T1	Utah Courts (<code>utcourts.gov</code> , <code>le.utah.gov</code>)	Two-source corroboration for Utah cases
T2	CourtListener API	Backup mirror (per <code>feedback_courtlistener_deprecation.md</code> , not preferred but available)
T3	Caselaw Access Project	Historical mirror
T4	Multi-AI Peer Review (Claude + Gemini consensus)	When T0-T3 all miss and the cite must be verified
AEGIS	Content-hash + containment check	Post-stream gate on any cited authority

The audit pipeline calls T0 first for every cite. If T0 misses, it escalates to T1 (the public Utah mirror) before declaring a cite unverifiable. If both miss, the cite is flagged for operator review — *never* silently passed.

6 · How the audit handles a real Hymas pass

The 2026-05-14 pass on the Hymas filing trio (URAP 10 motion, URAP 8 stay, TOA) produced:

- 47 case citations extracted from the three documents combined
- 47 corpus queries (one per cite)
- 3 cites with zero corpus matches → flagged → manual investigation found:
- Knowlton v. Knowlton 2023 UT App 16: FABRICATED → removed, replaced with Knowles + Bakanowski
- Knowlton v. Knowlton 2024 UT App 25: FABRICATED → removed

- Stillman v. Stillman 490 P.2d 326 (Utah 1971): NOT corpus-verifiable → removed, substance preserved via Dahl v. Dahl 2015 UT 23
- 1 cite with wrong neutral: Risher v. Emerson 2017 UT App 250 → corrected to 216 in 4 instances
- 1 cite with wrong subject: Wilkerson v. Stalder → removed entirely
- 12 rule citations extracted, all 12 verified against rule text in `rules_library`
- 1 rule subdivision wrong: URAP 4(b) → 4(c) in URAP 10 motion
- 1 rule reframed: URAP 23 page-limit framing → removed
- 1 rule attribution: URCP 7B meet-and-confer softening

After the audit, the documents were re-circulated through the Commission (paper 03) for unanimous reviewer sign-off. The result is preserved in `_UNANIMITY-CERTIFIED.md` and `_REVIEW-HYMAS-MOTIONS-FINAL.md`.

That cycle is the standing protocol. Every Hyman filing — every BenchSlap filing — goes through it.

7 · What the audit explicitly does not catch

The audit catches errors in citation, rule reference, and holding paraphrase. It does not catch:

- **Strategic errors** — citing a real case for a real holding that nonetheless argues against your position. The audit verifies the citation; the operator verifies the strategy.
- **Tone errors** — a paragraph that reads as petulant to a court. Subjective; the human reviewer's job, sometimes flagged by the Commission's unanimity pass.
- **Factual claims about the record** — "the trial court found X on page Y" — verifiable only against the actual record, which the audit does not have on hand unless explicitly loaded.
- **Procedural posture errors** — filing in the wrong court, the wrong docket, with the wrong caption. The audit checks the citations inside the document; it does not check the document's procedural envelope.

These remain operator responsibilities. The audit reduces the failure surface from "every citation, every rule, every holding" to a much smaller "every strategic call, every tone choice, every record-fact claim, every procedural posture." A real filing-quality gate is human + audit, not audit alone.

8 · One more pattern — the placeholder list

A subtle pre-filing failure class: brackets that were placeholders for the drafter to fill, that the drafter forgot to fill. Examples from past drafts:

- `[CLIENT NAME]` left in a caption block
- `<<<DATE>>>` left in a certificate of service
- `TODO: confirm party email` left in the conferral-letter draft

The audit greps the documents for `\[A-Z][^\]]\]`, `<<<. >>>`, `TODO`, `FIXME`, `XXX`, and produces a list. Every match is a pre-filing blocker. The operator clears the list; the filing is gated on zero open placeholders.

In the most recent Hymas audit (2026-05-15), the placeholder list was 8 entries (mostly OPC-supplement Dastan-name and Villalobos-caption fills). Each is flagged with file:line; each must be filled before the document is treated as filing-ready.

9 · One-line summary

A model's self-check on its own citations is no check at all. The pre-filing audit asks the corpus, not the model. Nine specific failure classes — fabrication, wrong neutral, hallucination at a real slot, wrong subject, overruled, wrong subdivision, reframed against text, attribution flip, cross-document inconsistency — are catchable by corpus and rule-text comparison. Every BenchSlap-drafted filing goes through it before submission. The filings that survived this audit in 2026 are the filings that did not get sanctioned.

*Authored 2026-05-15 by Richard L. Sanders, Utah Bar #15728. The nine pitfalls enumerated in §2 are each drawn from an actual pre-filing audit pass on a real BenchSlap-drafted document in 2026. The Hymas filing trio whose 2026-05-14 audit pass is referenced is preserved at `docs/client-matters/hymas/{08,09,10}.md` and its corresponding audit memo at `_VERIFICATION-RECORD-2026-05-14T001820.md`. Co-authored by Claude (Opus 4.7, 1M context).**