

AEGIS PRIME 2 — Holding-Binding-Weight Inference

Telling the load-bearing holding from the dicta, the rejected dictum, and the silently-overruled case.

Author: Richard L. Sanders · Utah Bar #15728 **Role:** Sole inventor and implementer
Version: 1.0 — 2026-05-15 **Companion papers:** `01-deterministic-closed-corpus-verification.md` (AEGIS — text containment) · `08-aegis-prime-structural-verification.md` (AEGIS PRIME — disposition / panel-vote / treatment graph) **Source:** `lib/case-fact-extractor.js`, `lib/binding-weight-inference.js`, `lib/treatment-graph.js`, `lib/post-stream-gate.js`, `scripts/extract-platinum.js`, `scripts/build-treatment-graph.js`, migration 179 columns `holding_binding_weight` + `superseded_by_opinion_id` + `case_treatment_graph`, in the public repository at <<https://github.com/Benchslap/Benchslap>>.

Abstract

The first AEGIS PRIME paper (08) introduced four structural columns on every opinion row: `disposition`, `panel_vote`, `holding_binding_weight`, and `case_treatment_graph`. It described the *schema* and the *hard-block matrix* that enforces it. It did not describe how the system decides, when a new opinion lands in the corpus, what value to put in the `holding_binding_weight` column — and how that value updates over time as later opinions overrule or distinguish the case.

That decision is the subject of this paper.

Binding-weight inference is not a single classification. It is a five-way enum (`BINDING`, `PERSUASIVE`, `DICTA`, `REJECTED`, `SUPERSEDED`) that must be inferred from a combination of (a) panel-composition signals inside the opinion itself, (b) language patterns that distinguish holdings from dicta, (c) jurisdiction-specific stare-decisis rules, and (d) downstream signals from the treatment graph as later opinions touch the case. The naïve approach — "majority opinion = BINDING, everything else = lower weight" — fails on enough cases to make the entire AEGIS PRIME hard-block matrix unreliable: when the gate blocks a citation as `BINDING_WEIGHT_MISMATCH` but the underlying classification was

wrong, the gate has produced a false positive, the user has lost trust, and the system has earned a reputation worse than not running the check at all.

This paper describes the multi-signal inference pipeline that produces the binding-weight column with a recall-precision profile high enough to support hard blocks in production. It also describes the temporal update mechanism: every overruling, supersession, or distinguishing reference in a newly-ingested opinion triggers a recomputation of the affected target's `holding_binding_weight` — so a case that was `BINDING` yesterday may be `SUPERSEDED` today, and the verify gate sees the new value the next time anyone cites the case.

1 · Why the naïve heuristic fails

A first cut at binding-weight inference assigns `BINDING` to every published majority opinion of the highest controlling court, `PERSUASIVE` to lower-court / out-of-jurisdiction opinions, and `DICTA` to anything else. That heuristic works for textbook hypotheticals. It fails on the case shapes that actually show up in production:

Single-issue split panels. *State v. Smith* is decided 4-3. The four-vote majority addresses Issue A; on Issue B, three of those four agree but the fourth refuses to reach Issue B as unnecessary to the disposition. The textual majority for Issue B is therefore 3 of 7 — not binding. A naïve heuristic that reads "majority opinion" as one monolithic block of binding holding misclassifies the Issue B passage.

Holding-by-implication. *State v. Jones* affirms a conviction "for the reasons stated in the State's brief." The State's brief is not in the corpus. The naïve heuristic reads the affirmance language as `BINDING` and the issue-level holdings as derivable from that language. They are not derivable. The opinion's binding-weight on any particular issue is `NOT_EXTRACTABLE`; it cannot be safely cited for any proposition the State's brief might have argued.

Dicta that look like holdings. *State v. Brown* upholds a conviction on Confrontation Clause grounds, and devotes ten paragraphs to a hypothetical analysis of what would happen if the State had introduced the disputed statement under a different exception. The hypothetical analysis is dicta. A naïve heuristic that classifies all reasoned-paragraphs-of-the-majority-opinion as `BINDING` produces a false positive on the dicta paragraphs.

Conditional holdings. *State v. Davis* adopts a four-part test for harmless-error review, then says "we need not apply the fourth prong here because the third prong is dispositive." The fourth prong was articulated but not applied. Some Utah cases later treat the fourth prong as binding (because the panel adopted it); some treat it as dicta (because the panel did not apply it). The correct binding-weight depends on a later opinion's choice — a temporal signal the initial inference cannot resolve.

Concurrences that become majorities. A two-justice concurrence in *State v. Wilson* is later quoted by the Utah Supreme Court in *State v. Yates* as the operative rule. *Wilson*'s concurrence was **PERSUASIVE** at ingest time, but as of the *Yates* citation it is the rule. Binding-weight must update.

The Pingree problem. Under Utah law (and most jurisdictions), the Utah Court of Appeals is bound by its own prior decisions — *Pingree v. Cossette*, 424 P.3d 1242 (Utah 2018), ¶ 15 (horizontal stare decisis). District courts within the Court of Appeals' territorial reach are bound by published Court of Appeals decisions — *Sterling Fiduciaries LLC v. JPMorgan Chase Bank*, 2017 UT App 135, ¶ 14. Binding-weight inference must therefore consider both **the citing jurisdiction** and **the cited jurisdiction**, not just the cited court's level. A Utah Court of Appeals decision is **BINDING** in a Utah district court and **PERSUASIVE** in a Nevada federal district court. The same opinion row has different binding-weight depending on the forum of the citing motion.

The naïve heuristic produces wrong answers on all six of these case shapes. In a corpus of approximately 250,000 published opinions across the closed jurisdictions BenchSlap supports, an error rate as low as 1% would produce ~2,500 wrong classifications — and at hard-block enforcement, ~2,500 false positives that erode user trust before the system has caught its first real misuse.

2 · The four signals

Binding-weight inference in AEGIS PRIME 2 reads four independent signal channels. The output enum is computed from a small rule engine over those four channels. The signals are deliberately chosen to be mostly orthogonal — different signals fail on different case shapes, so the joint reliability of the four-signal aggregate is materially higher than any single signal.

Signal 1 — Panel composition (intra-opinion)

Extracted at ingest time by **case-fact-extractor.js**'s regex pass, with overflow handled by the Platinum LLM pass (**scripts/extract-platinum.js**).

For each opinion, the extractor identifies:

- The author of the opinion (judge name string).
- The list of judges joining the majority (typically a "Judge X concurred in by Judges Y and Z" line).
- The list of judges concurring in the result but writing separately.
- The list of judges concurring in part and dissenting in part.
- The list of dissenters.

The signal is stored as `panel_vote` in the opinion row. It is used by the inference engine to compute, for any quoted passage in the opinion, how many judges joined it. A passage that bears the signature of 4 of 7 judges is on safer binding-weight ground than a passage that bears the signature of 2 of 7.

Signal 2 — Holding-vs-dicta language patterns (intra-opinion)

A set of regex + Platinum-LLM patterns flag passages that the opinion itself labels as one or the other:

- Holding labels: "We hold that ...", "The rule we adopt is ...", "Today we decide ...", "Our holding is ...", and similar.
- Dicta labels: "we need not decide ...", "the question is not before us ...", "even assuming arguendo ...", "by way of illustration ...", "the parties have not raised ...", and similar.
- Reservation labels: "we leave for another day ...", "we express no view on ...", "the matter is not properly before us ...".

When a passage in the opinion text is bracketed by holding language, the inference engine raises its candidate binding-weight one notch (toward `BINDING`). When bracketed by dicta or reservation language, it drops one notch (toward `DICTA`).

The patterns are deliberately conservative — high precision, lower recall. A passage neither bracketed nor obviously labeled defaults to the panel-composition signal alone. The engine never *invents* a holding label that the opinion did not use.

Signal 3 — Treatment-graph downstream signals (inter-opinion, temporal)

`scripts/build-treatment-graph.js` scans every newly-ingested opinion for explicit treatment of older opinions. The edge types are:

- `OVERRULED` — the new opinion says "*Smith v. Jones*, 2020 UT 5, is overruled." The target's binding-weight is forced to `SUPERSEDED` on the issues addressed; `superseded_by_opinion_id` is set to the source.
- `SUPERSEDED` — by statute or rule rather than judicial overruling. Same treatment.
- `DISTINGUISHED` — the new opinion narrows the target. The target's binding-weight stays `BINDING` for the narrowed scenario but the treatment-graph edge is preserved so downstream citing parties see the narrowing.
- `NEGATIVELY_CITED` — the new opinion criticizes the target without overruling. Binding-weight unchanged, but the citation will surface a `NEGATIVELY_CITED` flag at verify time so the drafter sees the negative treatment.
- `REJECTED` — the new opinion expressly rejects a specific holding within the target. The target's binding-weight on that specific holding drops to `REJECTED` (a level

below **DICTA** — a holding that the issuing court has since disclaimed).

The treatment graph is bidirectional: if A overrules B, A has an outgoing **OVERRULED → B** edge AND B has an incoming **OVERRULED_BY ← A** edge. The verify-time check at the binding-weight column reads the incoming edges on the cited case; if any incoming edge is **OVERRULED_BY**, the binding-weight is **SUPERSEDED** regardless of what the initial intra-opinion inference computed.

Signal 4 — Forum-of-citation (verify-time, contextual)

The first three signals fix a *base* binding-weight on the cited opinion. The fourth signal adjusts it for the **forum of the citing motion**:

- Same-court horizontal stare decisis (Utah COA citing Utah COA): **BINDING** under *Pingree / Sterling Fiduciaries*.
- Vertical stare decisis (district court citing Utah COA): **BINDING** under *Stewart v. Coffman*.
- Reverse vertical (Utah COA citing Utah district court): **PERSUASIVE** at most, often **NOT_BINDING**.
- Cross-jurisdiction (Utah court citing Nevada COA): **PERSUASIVE**.
- Foreign / unpublished: **PERSUASIVE** at best; some jurisdictions impose stricter limits on citation of unpublished opinions, which the engine applies based on local rule.

Forum information arrives at verify time from the document's silo-jurisdiction metadata (every BenchSlap silo carries a **jurisdiction** field). The verify path looks up the cited opinion's *base* binding-weight, then applies the forum overlay to produce the *effective* binding-weight for the specific motion being drafted. The hard-block matrix evaluates against the effective binding-weight, not the base.

3 · The rule engine

The four signals feed a small deterministic rule engine in **lib/binding-weight-inference.js**. The engine output is the *base* binding-weight for the cited opinion-passage pair; the verify path applies the forum overlay (Signal 4) at lookup time.

```
inferBaseBindingWeight(opinion, passage) {
  // Signal 3 dominates: treatment-graph wins over intra-opinion signals.
  if (opinion.case_treatment_graph.incoming.some(e => e.type ===
    'OVERRULED_BY' ||
    'SUPERSEDED_BY')) {
    return 'SUPERSEDED';
  }
}
```

```

    if (opinion.case_treatment_graph.incoming.some(e => e.type ===
'REJECTED_BY' &&
e.target_pinpoint_matches(passage))) {
        return 'REJECTED';
    }

    // Signal 2: explicit holding / dicta / reservation labels.
    const label = passageLanguageLabel(passage); // 'HOLDING' | 'DICTA' |
'RESERVATION' | null
    if (label === 'DICTA' || label === 'RESERVATION') return 'DICTA';

    // Signal 1: panel composition.
    const joined = countJudgesJoining(opinion, passage);
    const total = opinion.panel_vote.total;
    if (joined >= Math.ceil(total / 2)) {
        return label === 'HOLDING' ? 'BINDING' : 'BINDING'; // panel
majority + reasoned passage
    }
    if (joined >= 2 && total >= 5) {
        return 'PERSUASIVE'; // plurality / concurrence rather than
majority
    }

    return 'PERSUASIVE'; // safe default
}

```

The engine's defaults are conservative: when in doubt, drop a notch toward **PERSUASIVE**. The downstream cost of a false **BINDING** classification (a hard-block false-positive that frustrates the user) is materially higher than the cost of a false **PERSUASIVE** classification (the gate doesn't block; the user is responsible for their own argument-strength judgment). The engine optimizes for that asymmetry.

4 • The hard-block trigger

`lib/post-stream-gate.js` evaluates an AI-generated citation against the stored binding-weight using the forum-adjusted *effective* weight. The trigger is:

- **SUPERSEDED_CASE** **hard block**. If the cited opinion's effective binding-weight is **SUPERSEDED**, the gate replaces the citation with a notation pointing at the overruling opinion and blocks the surrounding sentence's claim. Substitute text: "[CITATION SUPERSEDED — see {overruling case}; original {cited case} was overruled on this point]". The AI is required to reissue the analysis citing the overruling opinion. No user override is offered: cite-checking a superseded case to a court is itself a Rule-11 problem the gate refuses to facilitate.

- **BINDING_WEIGHT_MISMATCH hard block.** If the AI's surrounding text frames a dicta-level holding as "controlling" or "binding," and the stored effective binding-weight is **DICTA**, the gate replaces the framing with accurate framing. Substitute text: "[CITATION FRAMED AS BINDING; actual binding-weight is DICTA / PERSUASIVE]". The AI is required to reissue. Soft override is offered (the user can confirm "I know it's dicta, file anyway") because there are legitimate reasons to cite dicta — but the framing has to change first.
- **REJECTED warning.** If the cited opinion's effective binding-weight is **REJECTED**, the gate issues a soft warning rather than a hard block. The drafter is told that the issuing court has since disclaimed the specific holding. Some appellate strategies legitimately cite rejected holdings (to distinguish, to ask for re-examination), so the warning lets the drafter proceed once acknowledged.

The three triggers compose with the existing AEGIS / AEGIS PRIME 1 triggers (text-containment, disposition mismatch, attribution mismatch) without conflict. A given citation can fail multiple triggers simultaneously; the gate emits one combined block message rather than a cascade.

5 • Update on overrule

The temporal signal — Signal 3 — is the architectural feature that distinguishes AEGIS PRIME 2 from a static classification system. The update flow is:

1. A new opinion lands in the corpus via the harvester.
2. `scripts/build-treatment-graph.js` runs immediately on the new opinion. It scans the text for treatment language (**overrule**, **supersede**, **reject**, **distinguish**, **disapprove**, **abrogate**) and resolves the targets against the `opinion_library` index.
3. For each target, an outgoing edge is added to the new opinion's `case_treatment_graph` AND an incoming edge is added to the target's `case_treatment_graph`. The two updates run in the same transaction.
4. The target's `holding_binding_weight` is recomputed using the inference engine (Section 3). For an **OVERRULED_BY** or **SUPERSEDED_BY** edge, the recomputation forces the value to **SUPERSEDED**. For a **REJECTED_BY** edge pointing at a specific pinpoint, only the binding-weight for that pinpoint range drops.
5. The `content_hash` of the target row is recomputed alongside the new column values (AEGIS drift-prevention trigger requires hash refresh on any text-bearing column change). The recomputation is atomic.
6. Any future verify call on a citation to the target sees the updated `holding_binding_weight` immediately. No cache invalidation is needed because

the verify path always reads the database directly.

The update flow runs on every overruling — including overrulings that the legal-publishing industry's commercial citators have not yet indexed. The closed-corpus / batch-scan approach can be 24-72 hours faster than commercial citators on newly-published overrulings, because the harvester runs daily and the treatment-graph builder runs on every new opinion in the same pipeline. The trade is breadth: commercial citators cover a wider universe; the BenchSlap closed corpus covers the jurisdictions where the user has chosen to drift the highest-precision check.

6 • A concrete case: *Stillman* and *Dahl*

A worked example from the production corpus illustrates the inference and update flow.

For decades, the leading Utah authority on the ability-to-pay requirement in domestic attorney-fee awards was *Stillman v. Stillman*, 490 P2d 326 (Utah 1971). It established a pattern of language that lower courts cited freely. AI drafters trained on appellate-practice text routinely quoted *Stillman* for the proposition that "the trial court must find that the spouse has the ability to pay before fees may be awarded."

In 2015 the Utah Supreme Court decided *Dahl v. Dahl*, 2015 UT 23, 345 P3d 566. The relevant holding is at ¶¶ 168–69. *Dahl* did not formally overrule *Stillman* — instead, it announced an articulated ability-to-pay standard that is functionally a complete restatement. Lower courts following *Dahl* sometimes still cite *Stillman* parenthetically; many do not.

When the corpus harvester ingested *Dahl* in 2015, the treatment-graph builder caught the phrase "we today reaffirm and elaborate on the rule articulated in *Stillman*" — and added a **REAFFIRMED_BY → Dahl** edge to *Stillman*. The binding-weight inference engine read that signal: an incoming **REAFFIRMED_BY** edge does not drop binding-weight; it raises confidence. *Stillman* remained **BINDING** in the corpus.

Then, in the audit that preceded the 2026-05-15 Hymas filing, a forensic cite-check discovered that the AEGIS-containment layer was giving *Stillman* citations GREEN flags on quoted text that did not actually appear in the *Stillman* opinion. The phrase being quoted ("the trial court must find ability to pay") was *Dahl*'s rewording, not *Stillman*'s text. The drafter had been silently substituting one source for another for years across the corpus — 47 occurrences in three filing-ready documents alone. None of those 47 occurrences were detected by the text-containment layer because the *form* of the citation (case name + reporter cite) matched a real opinion and AEGIS does not check whether the case name matches the cited proposition.

AEGIS PRIME 2's response was not to overrule *Stillman* — *Stillman* remains a real case — but to add a **FABRICATED_QUOTE → Stillman** flag at the pinpoint level. The next time the

drafter citation surface tries to cite *Stillman* for the disputed proposition, the binding-weight column returns `REJECTED` for that pinpoint range (the issuing court has since articulated the rule differently in *Dahl*, and AI-generated quotes attributed to *Stillman* are no longer reliable). The substitute path points at *Dahl*'s actual paragraph numbers. The drafter is required to re-cite or to acknowledge the change.

This is the temporal-update mechanism doing exactly what it was built to do: not "is *Stillman* still good law" (it is) but "is the AI-quoted proposition actually in *Stillman*" (it was not, and now we know).

7 · Reproducibility

The binding-weight inference engine, the rule engine, the treatment-graph builder, and the gate trigger are all in the open-source BenchSlap repository:

- Schema: migration `179` (`opinion_library.holding_binding_weight`, `superseded_by_opinion_id`, `case_treatment_graph`).
- Extraction: `lib/case-fact-extractor.js` (regex-first) + `scripts/extract-platinum.js` (LLM second pass).
- Inference: `lib/binding-weight-inference.js`.
- Treatment graph: `scripts/build-treatment-graph.js` + `lib/treatment-graph.js`.
- Gate: `lib/post-stream-gate.js` (sections `evaluateBindingWeightTrigger` and `evaluateSupersededCaseTrigger`).
- Demo: `POST /api/admin/aegis-prime/verify` with a `claim` field that references binding-weight; `GET /api/admin/aegis-prime/binding-weight/:opinion_id` returns the stored value.

A test fixture covering the six failure-mode case shapes from Section 1 (split panel, holding-by-implication, dicta-that-look-like-holdings, conditional holdings, concurrences-that-become-majorities, the Pingree-forum problem) is in `tests/aegis-prime-binding-weight.test.js`.

The treatment-graph build is incremental: on each harvester run, only new opinions are scanned for outgoing treatment edges, and only the targets they touch get their `holding_binding_weight` recomputed. A full rebuild — useful for after a schema change or after correcting a misclassification — runs in approximately 8 minutes on the production corpus (250k opinions).

8 · Novelty claims

1. **Multi-signal binding-weight inference for judicial opinions**, in which a small deterministic rule engine combines panel-composition signals, intra-opinion holding-vs-dicta language patterns, treatment-graph downstream edges, and forum-of-citation overlays to produce a five-way enum that is precise enough to support hard-block enforcement at the AI-output gate.
1. **Bidirectional treatment-graph updates triggered by every new opinion ingest**, so the binding-weight of any cited case reflects the current state of the law within 24 hours of an overruling rather than the 1-6 weeks typical of commercial citator latency.
1. **Pinpoint-level rejected-holding granularity**, in which a **REJECTED_BY** edge can target a specific pinpoint range within an opinion rather than the opinion as a whole, so the binding-weight column can carry "still binding for proposition A, rejected for proposition B" without losing the rest of the opinion to a single coarse classification.
1. **Forum-overlay binding-weight resolution at verify time**, in which the same opinion row exposes different effective binding-weights depending on the jurisdiction of the citing motion, eliminating the false-positive class produced by classification systems that treat binding-weight as a single global value.

The four elements together are AEGIS PRIME 2. Companion paper 08 covers the schema and the hard-block matrix; this paper covers the inference path that produces the value in the column. Together with AEGIS (paper 01, text containment) and AEGIS PRIME 1 (disposition / panel-vote / treatment-graph), they form the verification gate that prevents the four classes of legal-AI hallucination cited courts cannot detect from text-containment alone: (a) put-words-in-the-court's-mouth, (b) get-the-disposition-wrong, (c) attribute-the-dissent-as-majority, (d) cite-the-overruled-case-as-good-law.