

CGFEV: Auditable LLM-Assisted Qualification of Configuration-Sensitive Manufacturing Evidence for Commercial Seating Design

Xu Huang¹, Chuanzhi Chen^{1,*}

¹Qilu University of Technology, Jinan 250300, Shandong, China

Abstract

Engineering rules often rely on dimensions copied from supplier documents. A value may look correct yet belong to another chair option, measurement object, or source that lacks a required condition. Contract-Guided Frozen-Evidence Verification (CGFEV) freezes assigned files and target configuration, while a Rule Card states what must be established. An LLM writes typed propositions and locators; deterministic Gate v0.2 controls release. We evaluated four frozen routes on commercial seating. Holdout-A contained 48 family-isolated natural cases, including 47 with binary Gold; Holdout-B contained 20 controlled pairs. On Holdout-A, B1 released 4/8 Gold-ELIGIBLE cases and falsely admitted 6/39 Gold-NOT_ELIGIBLE cases. R0, L1, and the text-only V1 prompt variant recorded no observed false admissions and released 0/8 eligible cases. R0 completed 47/47 binary cases, whereas L1 and V1 ended in ERROR on 12 and 14 cases, leaving substantive decisions for 35/47 and 33/47. Every route missed all six eligible members and all four required direction changes in Holdout-B. All 88 formal V1 records had empty visual assets fields, so these results provide no evidence about visual or multimodal qualification. Together, these results distinguish false admission, eligible release, and execution completion as separate evaluation dimensions. CGFEV provides an auditable admission boundary for configuration-sensitive evidence qualification.

Keywords: evidence qualification, product configuration, large language models, intelligent manufacturing, automated compliance checking, provenance

Received on 19 August 2026, accepted on 11 September 2026, published on 09 October 2026

Copyright © 2026 Xu Huang *et al.*, licensed to EAI. This is an open access article distributed under the terms of the [CC BY-NC-SA 4.0](#), which permits copying, redistributing, remixing, transformation, and building upon the material in any medium so long as the original work is properly cited.

doi: 10.4108/eetsis.14612

1. Introduction

A value reported in a supplier document is not necessarily ready for engineering use simply because its magnitude and unit appear correct. It may describe a different chair, another configuration of the same chair, or a dimension taken from the wrong place. Studies of product configuration represent parts, attributes, relations, constraints, and requirements [1-4]. Data-sheet extraction can then move reported values into an engineering record [5]. Between these two steps, someone

still has to decide whether this value can be used for the rule and configuration now under review.

The current document processing methods excel at identifying candidate facts and can locate numerical values, units, product names, and their surrounding text [6-8]. Large model systems can extract and normalize product attributes [9-10]; layout-aware models, OCR-free models, and table verification methods provide alternative information parsing paths for the same batch of documents [11-13]. However,

*Corresponding author. Email: chenchuanzhi123@126.com

none of the individual steps alone can confirm that the extracted numerical values belong to the specified seat configuration version in the rules.

We did not treat an extracted value as final. In each case, we kept its product name, configuration, measured object, document version, and locator beside it. Breaking any of those links stopped the value from reaching the rule.

This extra step arose from the supplier files themselves, which were not written for the compliance system [14-15]. One file could cover several chair options, and a drawing might omit the condition that limits a dimension to one of them. A plausible number was therefore not enough to start the threshold check.

CGFEV records this intermediate decision for one rule and one stated configuration. The Rule Card fixes what must be checked before execution: identity, measurement meaning, conditions, source scope, conflict handling, and locators. The

saved record shows the check that passed the evidence onward or stopped it.

Research question. Can an LLM read the evidence fixed for one rule and one exact configuration while code keeps authority over release? We examine both useful release and admission of evidence with a known qualification problem. A completed rejection is not treated as equivalent to ABSTAIN or ERROR, and the paired cases are used to see whether a change in one condition moves the route in the required direction.

For every case, we froze the files before running a route. The Rule Card had already specified the target and the evidence checks. The LLM could read the assigned material and return a value with its location, but it could neither substitute another source nor change the target. It also had no release authority. Gate v0.2 checked the returned record, and the saved log retained the quotations, versions, failed checks, and final route (Figures 1 and 2)

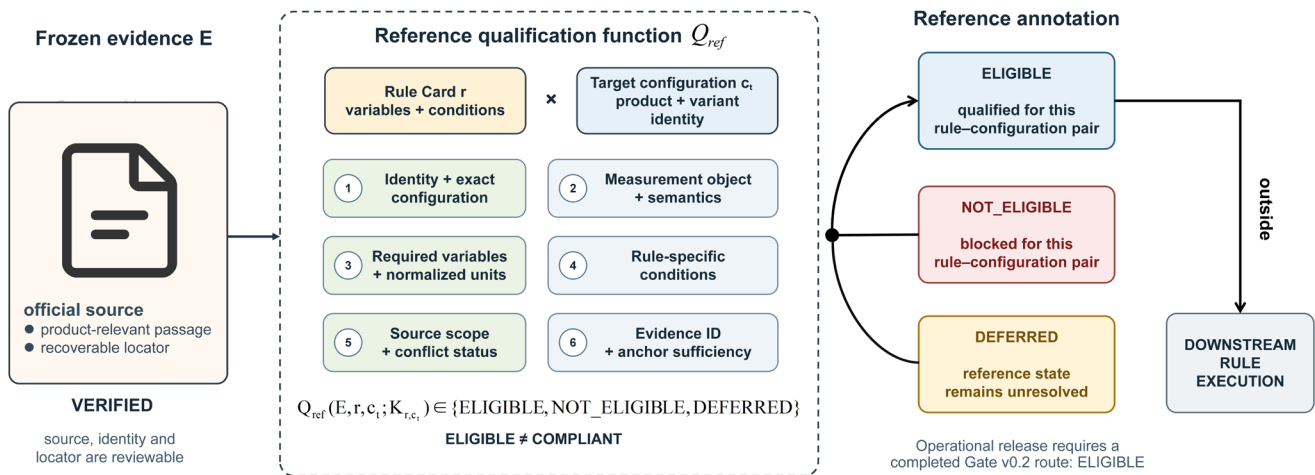


Figure 1. Evidence is checked before the selected engineering rule is allowed to run

Reference qualification and executable routing are distinct: Q_{ref} may be ELIGIBLE, NOT_ELIGIBLE, or DEFERRED, whereas Gate v0.2 issues ADMIT or REJECT for a completed package. Source verification and downstream compliance remain outside this Gate.

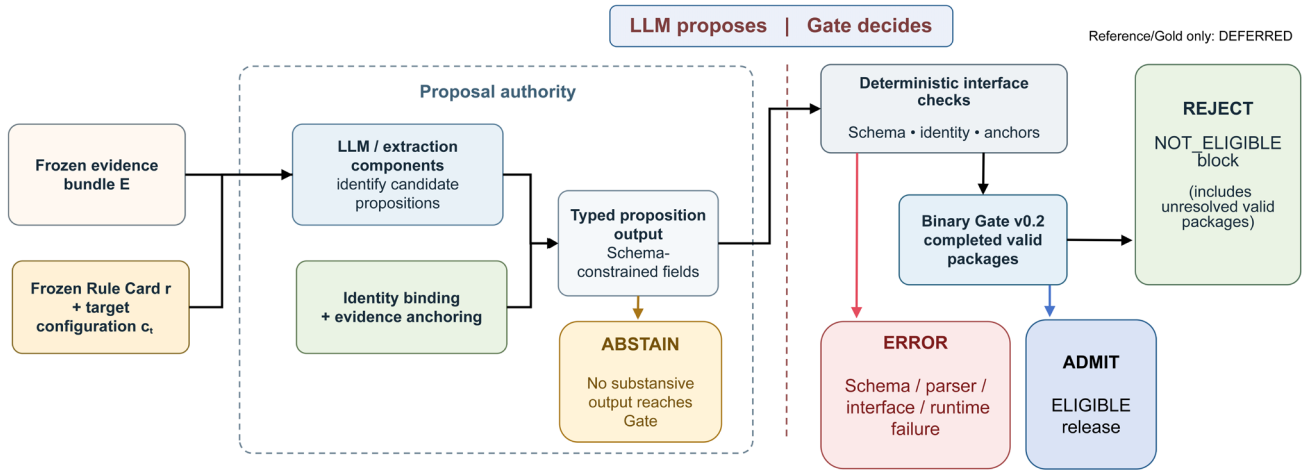


Figure 2. Proposition generation and deterministic routing

The proposal component reads only assigned evidence. Gate v0.2 archives ADMIT (ELIGIBLE) or REJECT (NOT_ELIGIBLE) for completed packages; ABSTAIN and ERROR remain runtime outcomes, and DEFERRED remains a Gold state.

We count a case at the bundle-rule-configuration level. Finding the source happens earlier; deciding product compliance happens later.

A run may also stop before Gate v0.2 acts. The log then records ABSTAIN or ERROR, while Gold keeps ELIGIBLE, NOT_ELIGIBLE, and DEFERRED as its reference states.

For a completed package, the Rule Card supplies the checks, the proposal component supplies fields and locators, and Gate v0.2 alone controls release to the rule executor.

We tested this arrangement with product families kept outside Development and with pairs that changed one qualification condition. Independently prepared Gold supported separate counts for release, false admission, completion, substantive decisions, and required pair direction.

The observed failures were not interchangeable. B1 released some disqualified evidence; R0 usually completed but withheld eligible evidence; L1 and V1 also lost cases to ERROR. The 16-case identity audit remains Development-only, and no experiment here tests seat safety, downstream compliance, or another manufacturing domain.

2. Related Work

Automated compliance checking usually assumes that product information has already been structured [16-19]. The supplier files used here precede that point. They can place several options in one document or put the needed dimension beside a similar but inapplicable one. A rule can test the field only after that ambiguity has been resolved.

Work on evidence support addresses part of this difficulty. RAG links generated answers to retrieved material [20], and claim-verification research covers fixed, changing, and conflicting evidence [21-23]. Citation and RAG evaluations ask whether an answer is supported [24-25], while hallucination studies examine unsupported output [26].

Reject options, confidence calibration, and dataset-shift research concern whether a system should answer at all [27-30]. These approaches do not normally test whether a source value is bound to a particular manufactured variant.

Selective classification determines whether a predictor should return a label under uncertainty [27-30]. RAG and claim-verification methods test whether an answer is supported by retrieved evidence [20-26]. Attribute extraction recovers values [8-10, 31], while schema constraints govern the form of the returned record [32-33]. CGFEV addresses a different handoff: it asks whether a located value is admissible for one predeclared engineering rule and exact product configuration. The proposal stage can construct the record, but the deterministic Gate retains release authority.

A provenance record can preserve where an artifact came from, but origin is not the same as fitness for a present rule. Datasheets and model cards document production, intended use, and known limits; PROV-DM provides a general provenance representation [34-36]. An engineer still has to decide whether the recorded artifact supports the configuration now being checked.

That decision is closer to the concerns of product-configuration and data-quality research. Configuration models represent products through parts, attributes, constraints, and requirements [37]. Data quality also depends on intended use, and prior work distinguishes properties of the data from the context in which they are used [38-39]. In this study, context has a specific operational meaning: whether a dimension can be tied to the chair option under review [40].

LLMs can read attributes and normalize values [31], while grammars and templates constrain the returned form [32-33]. In CGFEV, these tools prepare a candidate record. They do not decide whether it belongs to the target configuration; a structurally valid record may still fail identity, meaning, condition, source-scope, conflict, or anchor checks.

Adjacent systems constrain different points in the

authority boundary. NeMo Guardrails interposes a programmable runtime and can invoke developer-defined execution rails [41]. PICARD uses incremental parsing to block token continuations that fail lexical, grammatical, logical, or semantic checks in text-to-SQL generation [42]. IsolateGPT isolates third-party app execution and mediates cross-component interactions [43]. These mechanisms constrain dialogue, generated structure, or downstream action. CGFEV also checks rule and configuration-specific evidence admissibility before its deterministic Gate permits rule execution. A schema-valid record can still fail identity, configuration, meaning, condition, source scope, conflict, or anchor checks.

These studies address nearby parts of the problem. Some separate prepared information from rule execution [44], mark claims when evidence is insufficient [45], or formalize supplier statements [46]. Traceability work connects requirements to development records [47]. Requirements research also examines language, semantic links, and contextual conflict [48-50], while anthropometric evidence depends on population and setting [51]. We place these concerns at the admission point immediately before one specified engineering-rule run.

Furniture-specific studies show how customer-defined dimensions propagate into 3D models, bills of materials, technical drawings, and fabrication outputs [52]. A furniture digital-product-passport prototype frames lifecycle traceability around data governance, interoperability, and integrity [53]. Neither study evaluates commercial-seating compliance evidence. We use the manufacturing traceability review as a bridge: product identifiers, attribute records, and process records connect physical products to digital data and vary by implementation setting [54]. CGFEV applies this context to one frozen evidence package, seating rule, and exact product configuration.

3. Methods

We first freeze the evidence, target configuration, and qualification contract. We then evaluate four routes: B1 serves as the field-presence baseline, R0 applies deterministic extraction rules, and L1 uses a text model to populate typed proposition records. V1 remains the frozen route identifier for audit continuity and is called the text-only V1 prompt variant. All 88 formal V1 records had an empty `visual_assets` field. The formal evaluation therefore provides no evidence about visual or multimodal evidence processing. R0, L1, and V1 use the same 13-field Schema. Gate v0.2 routes each completed package. Runs that produce no such route are recorded as ABSTAIN or ERROR. Figure 1 separates these runtime outcomes from the three Gold states.

3.1. Qualification separates reference states from executable routes

Let $E = \{e_1, \dots, e_n\}$ be the frozen evidence bundle, r the engineering rule, c_t the target product configuration, and K_{r,c_t} the contract for this rule and configuration. Every evidence item has an ID, source metadata, a hash, and a text or region locator. Once E is frozen, the run cannot add, replace, repair, or supplement any item. It also cannot change the target or use general product knowledge. K_{r,c_t} lists the variables, identity and configuration links, measurement meaning, required conditions, source scope, conflict rule, and anchor rule.

The task-level reference decision is

$$Q_{\text{ref}}(E, r, c_t; K_{r,c_t}) \in \{\text{ELIGIBLE}, \text{NOT_ELIGIBLE}, \text{DEFERRED}\} \quad (1)$$

The reference annotation assigns one of three states. ELIGIBLE means that the frozen record satisfies every contract condition. NOT_ELIGIBLE identifies a disqualifying violation, and DEFERRED records a case that the available evidence cannot resolve. Both latter states block execution, but they call for different forms of review.

Plain language: Gold-ELIGIBLE is admissible for the rule and exact configuration. Gold-NOT_ELIGIBLE records a disqualifying condition established by the reference evidence. Gold-DEFERRED means that the frozen evidence cannot resolve admissibility.

Gate v0.2 implements a narrower binary route for completed packages: ELIGIBLE or NOT_ELIGIBLE. The method status is recorded separately as COMPLETED, ABSTAIN, or ERROR. ABSTAIN supplies no route, and ERROR identifies execution, parsing, Schema, evidence-reference, or interface failure.

Plain language: COMPLETED means that the method produced a Gate route. ABSTAIN means that it returned no substantive route. ERROR means that execution or an interface failure prevented a route.

DEFERRED concerns the reference annotation: the frozen evidence does not let annotators resolve qualification for the named rule and configuration. ABSTAIN concerns runtime execution: the evaluated method returned no substantive Gate route for the case. We record Gold state and method outcome on separate axes.

$$A(G) = \text{ADMIT} \quad \text{if} \quad G = \text{ELIGIBLE} \quad (2)$$

$$A(G) = \text{REJECT} \quad \text{if} \quad G = \text{NOT_ELIGIBLE} \quad (3)$$

Table 1 separates qualification from assurance and compliance. VERIFIED confirms that a source can be identified and reopened. ELIGIBLE applies only to the named qualification contract. COMPLIANT is a later conclusion produced by rule execution. A verified source may therefore support any qualification state, and eligible evidence does not establish product compliance.

Table 1. CGFEV qualification states and semantic boundaries

Term	Semantic layer	Definition in CGFEV	Operational consequence	Must not be interpreted as
VERIFIED	Evidence assurance	The evidence source, target identity, and locator are sufficiently resolved for audit and review.	The evidence may be inspected and cited, but it must still be evaluated against the frozen rule-configuration contract.	QUALIFIED / ELIGIBLE or COMPLIANT
QUALIFIED / ELIGIBLE	Qualification outcome	The frozen evidence satisfies the applicable rule-configuration qualification contract. ELIGIBLE is the formal output label.	The evidence is eligible to enter downstream rule execution for the specified rule and target configuration.	VERIFIED alone or COMPLIANT
Gold-NOT_ELIGIBLE	Qualification reference state	The reference evidence establishes a definite mismatch, disallowed substitution, or unmet contract condition for the specified rule-configuration pair.	The case is assigned a non-admitting Gold reference label for binary evaluation.	Gate REJECT, ABSTAIN, or ERROR
Gate REJECT	Executable Gate action	The non-admitting route for a completed package with at least one failed or unresolved required Gate check.	Downstream rule execution is withheld; failed conditions remain available for mismatch handling or evidence escalation.	Gold-NOT_ELIGIBLE, ABSTAIN, or ERROR
DEFERRED	Qualification reference state	The frozen evidence and contract do not support a definitive ELIGIBLE or NOT_ELIGIBLE reference determination.	A Gold-DEFERRED reference label is assigned; no binary ELIGIBLE or NOT_ELIGIBLE determination is made, and the case remains separate from binary denominators where the frozen protocol specifies.	ABSTAIN
ABSTAIN	Method-run behavior	The method returns no substantive qualification decision for the instance.	Non-coverage blocks downstream execution without producing a route label and does not change the reference qualification state.	DEFERRED or ERROR
ERROR	Execution / interface outcome	Execution, Schema validation, parsing, or interface processing fails.	The failure is retained and counted separately; it is not deleted or converted into a substantive qualification decision.	NOT_ELIGIBLE or ABSTAIN
COMPLIANT	Downstream rule outcome	A product satisfies the applicable engineering rule after qualified evidence has entered downstream rule execution.	This conclusion is produced only after evidence qualification and lies outside the output space of $\mathcal{Q}_{ref}$	VERIFIED or ELIGIBLE

Note. Q_{ref} returns only the three qualification reference states shown in the formula. VERIFIED is an evidence-assurance status; ABSTAIN and ERROR are method-run outcomes recorded separately; COMPLIANT is a downstream rule-execution conclusion. Evidence basis. Frozen terminology and formalism register; frozen evaluation/statistical protocol; frozen ERROR/ABSTAIN scoring policy; PI-adjudicated contribution and claim registers.

3.2. Rule Cards make qualification conditions explicit

A Rule Card is the executable form of K_{r,c_t} . It states the measurement object, variables, units, exact-configuration rule, allowed measurement meaning, required conditions, forbidden substitutes, source and conflict rules, and acceptable anchors. Table 2 lists these items for the four seating rules: adjustable seat depth (CAND-ASD-01-v0.2), seat-height range (CAND-SH-01-v0.2), seat width (CAND-SW-01-v0.2), and armrest height relative to the seat (PRC-AH-01-v0.4).

Let P be the validated propositions for one sample, and let $b_j(P, c_t; K_{r,c_t})$ be the j -th Gate v0.2 check. Each check returns PASS or FAIL. A missing or unresolved required value returns FAIL. The executable Gate does not keep UNRESOLVED as a third route.

$$b_j(P, c_t; K_{r,c_t}) \in \{\text{PASS}, \text{FAIL}\} \quad (4)$$

Let $F(P, K_{r,c_t})$ collect the failed checks. The frozen aggregation is

$$F(P, K_{r,c_t}) = \{j : b_j(P, c_t; K_{r,c_t}) = \text{FAIL}\} \quad (5)$$

$$G_{v0.2}(P, K_{r,c_t}) = \text{ELIGIBLE} \quad \text{if} \quad F(P, K_{r,c_t}) = \emptyset \quad (6)$$

$$G_{v0.2}(P, K_{r,c_t}) = \text{NOT_ELIGIBLE} \quad \text{if} \quad F(P, K_{r,c_t}) \neq \emptyset \quad (7)$$

A measurement cannot approve its own use. The Gate compares every required proposition with the target identity, configuration, variable, unit, measurement meaning, source scope, conflict state, required conditions, and anchor. Gate v0.2 returns NOT_ELIGIBLE for either a clear mismatch or an unresolved requirement. The proposition record keeps the reason, while Gold can still use DEFERRED. The Rule Cards also block close but wrong substitutes. Examples include overall width instead of usable seat width, arm height from the floor instead of from the seat, and product height instead of seat height.

Every seating card expects millimeters and an exact target configuration (EXACT_CONFIGURATION). The evidence needed after that depends on the rule. Adjustable seat depth needs both endpoints, the named mechanism, and support for stable selectable positions. The two height rules also need both endpoints; seat width may use one value or a source-defined range. An alias is usable only when its mapping was fixed before execution. If the model supplies a different target, variable, or measurement object, the runner records the mismatch instead of correcting it.

Table 2. Items checked by the four seating Rule Cards

Rule Card	Required variables	Identity / configuration	Measurement object / semantics	Necessary conditions	Disallowed substitution or failure	Source, conflict, and anchor
CAND-ASD-01-v0.2	Seat-depth minimum and maximum endpoints; normalized units.	Exact target product and configuration; mechanism identity resolved.	Seat-depth endpoints; EXPLICIT or INFERRED_COMPATIBLE semantics.	Stable selectable position confirmed; stability and mechanism propositions anchored.	Unresolved/blank mechanism identity, unconfirmed stable position, or missing endpoint.	Confirmed source scope; no unresolved conflict; valid evidence IDs and continuous spans/regions.
CAND-SH-01-v0.2	Seat-height minimum and maximum endpoints; normalized units.	Exact target product and exact configuration.	Seat-height endpoints with seat-height semantics; EXPLICIT or INFERRED_COMPATIBLE.	Both endpoints must be present and bound to the target configuration.	A fixed scalar cannot be expanded into an unstated minimum/maximum interval.	Confirmed source scope; no unresolved conflict; valid evidence IDs and continuous spans/regions.

Rule Card	Required variables	Identity / configuration	Measurement object / semantics	Necessary conditions	Disallowed substitution or failure	Source, conflict, and anchor
CAND-SW-01-v0.2	Seat-width scalar or explicitly declared interval; normalized units.	Exact target product and exact configuration.	Seat width, not a neighboring width concept; EXPLICIT or INFERRED_C OMPATIBLE.	The scalar or interval must be explicitly attributable to the target configuration.	Overall width, arm-to-arm width, and back width are not substitutes.	Confirmed source scope; no unresolved conflict; valid evidence IDs and continuous spans/regions.
PRC-AH-01-v0.4	Armrest-height-relative-to-seat minimum and maximum endpoints; normalized units.	Exact target product and exact configuration.	Height relative to the seat; EXPLICIT or frozen-rule-compatible inference.	INFERRED_C OMPATIBLE is permitted only under the frozen Rule Card.	Floor-relative arm height is not an admissible substitute.	Confirmed source scope; no unresolved conflict; valid evidence IDs and continuous spans/regions.

Note. All four cards check the product and configuration, the measurement meaning, required variables and units, use conditions, source scope, conflicts, and anchors. Failure of any common or rule-specific item blocks the evidence for that card. This block concerns evidence use and does not state whether the chair passes the engineering rule.

Evidence basis. We used the frozen Rule Cards, Gate specification, and sampling and annotation protocol.

3.3. Authority separation constrains LLM assistance

Figure 2 shows six parts of the pipeline. The case begins with fixed E , r , c_t , its Rule Card, and a list of evidence IDs. R0, L1, and V1 can use only these items to form propositions. Before the package reaches Gate v0.2, the interface checks its Schema, target binding, evidence references, and anchors. We save the input, method version, request, raw response, parsed propositions, failed checks, route, runtime status, and seal. The input owns the product, configuration, rule, Rule Card version, and evidence set. A component may fill the value, unit, meaning, binding, scope, conflict, condition, and source location. It cannot change the target, fetch another file, edit the Rule Card, or decide compliance. After these checks, Gate v0.2 routes packages from R0, L1, and V1. B1 does not build the same package; it routes its field-presence test directly. For an LLM route, the model supplies the record and the Gate owns release.

$$\begin{aligned}
 P_i^{(m)} &= \Phi_m(S_i, r, c_i; K_{r,c_i}) = \{p_{i,j}^{(m)} \mid v_j \in V_r\}, \\
 m &\in \{R0, L1, V1\}; \\
 S_i &= \text{Serialize}(E_i, r, c_i; K_{r,c_i}).
 \end{aligned} \tag{8}$$

Equation (8) fixes the package for sample i : assigned evidence, Rule Card, preset target, and the rule-target contract. The card determines how many propositions the method must return, and cited evidence IDs must come from that sample. B1 is outside this interface because it routes a field match directly. R0, L1, and V1 use the 13-field Schema. When meaning, binding, scope, conflict, or a rule condition cannot be established, the corresponding field remains null and the record explains the unresolved item.

Algorithm 1. Build a proposition package and validate its interface

```

INPUTS: assigned evidence  $E_i$ ; Rule Card  $r$ ;
fixed target  $c_t$ ; route  $m$ ; allowed evidence
IDs  $A_i$ 
RETURNS: method_result plus {COMPLETED,
ABSTAIN, or ERROR}
1  $S_i \leftarrow \text{SERIALIZE}(E_i, r, c_t)$  using CGFEV-
EVIDENCE-SERIALIZER-v0.1-HO
2 if  $m = B1$ : test the rule term and a
supported numeric unit in serialized text;
return its direct route
3 if  $m = R0$ : scan each sorted evidence text
with the frozen rule-local patterns
4 normalize matched mm, cm or in values to
mm; retain the evidence_id and matched anchor
5 assign records in canonical variable order;
set exact binding/source scope from target-
token matches
6 derive conflict status from distinct
normalized candidates; apply ASD stability
and mechanism predicates
7 if  $m$  in {L1, V1}: issue one frozen prompt
call with  $r$ ,  $c_t$ ,  $S_i$  and Schema
8 if the frozen runtime marks method_status =
ABSTAIN: return (empty, ABSTAIN)
9 remove optional Markdown fences and parse
the first complete JSON array/object once
10 if parsing fails: return (empty, ERROR)
11 validate exact required-variable set,
exactly one record per variable, all 13
fields, enums, values,  $A_i$  and anchors
12 if validation fails: return (empty,
ERROR); do not repair, impute or make a
second model call
13 return ( $P_i$ , COMPLETED) to the
deterministic Gate v0.2

```

Model confidence is not used for admission. An unresolved field fails the Gate. An invalid Schema, evidence reference, or anchor instead ends the run with ERROR. The stored predicate names show why the route failed or why the run stopped.

3.4. Frozen serialization produces auditable propositions

The serializer writes evidence blocks in stable evidence-ID order. A block includes the source type, original filename, SHA-256 value, source or archive locator, available version or date, frozen page or region locator, and extracted text. Short files are included in full. For longer items, $W(e)$ constructs $B(e)$ from a fixed window order based on the Rule Card and declared product/configuration tokens. The serializer excludes Gold annotations, coordinator comments, later OCR corrections, and manual transcription. It emits [NO_TEXT_AVAILABLE] for pages without usable text and keeps visual material linked to its original file and bounded location.

$$S_i = H_0(r, c_i; K_{r,c_i}) \parallel \text{concat}_{e \in \text{sort}(E_i, \text{evidence_id})} B(e), \quad (9)$$

$$W(e) = T(e) \parallel L(e) \parallel R(e) \parallel O(e).$$

The preface records RULE_ID, the Rule Card, TARGET_PRODUCT, TARGET_CONFIGURATION, and EXACT_CONFIGURATION. For long documents, the serializer considers target/configuration windows $T(e)$, locator windows $L(e)$, rule-keyword windows $R(e)$, and remaining pages $O(e)$, in that order. L1 and V1 use the same ordering. Provider-limit truncation follows it as well and is logged for each evidence item. Algorithm 2 specifies the complete procedure.

Algorithm 2. Frozen evidence serialization

```

INPUT  $E_i$ , Rule Card  $r$ , exact target  $c_t$ ,
assigned evidence assets
OUTPUT serialized text  $S_i$  and a per-evidence
truncation log
1 write the fixed preface  $H_0(r, c_t, K_{r,c_t})$ 
2 sort evidence blocks by ascending stable
evidence_id
3 for each block  $e$ , write metadata fields,
frozen locator and source content
4 if  $e$  is short, retain all extracted text;
otherwise select  $T(e)$ , then  $L(e)$ , then  $R(e)$ ,
then  $O(e)$ 
5 write [NO_TEXT_AVAILABLE] when a renderable
page has no usable text
6 if provider limits require truncation,
apply the same priority and set truncation[ $e$ ]
= true
7 for V1, append only the original frozen
page/image assets already assigned to  $e$ 
8 return blocks joined by the frozen
delimiter --- and retain the serialization
manifest

```

The Schema creates one record for every variable named in the Rule Card. A range needs two records because it has two endpoints. Each record has 13 fields: target product, target configuration, variable, normalized value, unit, measurement meaning, configuration binding, source scope, conflict status, stable-position status, mechanism identity, evidence ID, and a source span or bounded area. Fixed categories are used when meaning, binding, scope, conflict, or a rule condition cannot be resolved. Product and configuration are copied from the input only to check consistency; they cannot replace c_t . If the source gives no supported value, the numeric field remains null. We did not put a Gold label, compliance answer, repair state, or model-selected target in this Schema.

For text, support_span has to copy one uninterrupted passage from the source. A table, drawing, or image uses a page, cell, or bounded area that can be opened later. The wrapper checks that the evidence ID belongs to the case and that an anchor was supplied. It did not compare every formal text anchor with the source one character at a time. The parser removes an optional Markdown fence and then makes one attempt. It returns ERROR for a parse failure, a category outside the Schema, an evidence ID not assigned to the case, or a bad anchor. No second attempt is made, and the runner does not fill a missing value.

3.5. The deterministic gate implements a binary fail-closed route

The interface passes a package to Gate v0.2 only after structural validation. The target comes from the sealed input, not from the package. Gate v0.2 has no access to the supplier file, Gold, model confidence, or the later compliance threshold. Variables, required fields, evidence IDs, and anchors are checked by the wrapper first; failure there is recorded as ERROR.

On an accepted package, Gate v0.2 evaluates identity and the Rule Card conditions, including variables, units, meaning, binding, scope, conflicts, rule-specific conditions, evidence assignment, and anchors. Any missing, failed, or unresolved item is written to failed_conditions. An empty list produces ADMIT; otherwise the completed package is REJECTED.

$$G_{v0.2}(P_i, r, c_i) = \text{ELIGIBLE} \quad \text{iff} \quad \forall c \in C(r),$$

$$c(P_i) = \text{PASS};$$

$$\text{otherwise NOT_ELIGIBLE,} \quad (10)$$

$$\text{for run}_{\text{status}} = \text{COMPLETED}$$

Gate v0.2 checklist

```

C1 run_status = COMPLETED
C2 target_product exactly equals the
predeclared target product
C3 target_configuration exactly equals the
predeclared target configuration
C4 every rule-required variable has one
record and a non-null normalized_value

```

```

C5 every value uses the Rule Card unit mm
after deterministic conversion
C6 measurement_semantics in {EXPLICIT,
INFERRED_COMPATIBLE}
C7 configuration_binding =
BOUND_TO_EXACT_TARGET
C8 source_scope = CONFIRMED and
conflict_status = NO_CONFLICT
C9 every evidence_id is assigned to the
sample and every anchor is non-empty
C10 ASD only: stable_position_status =
CONFIRMED, mechanism_identity resolved and
both are anchored
C11 non-ASD: stable_position_status =
NOT_APPLICABLE and mechanism_identity =
NOT_APPLICABLE
C12 any failed condition is retained in
failed_conditions and blocks the completed
route

```

Algorithm 3. Deterministic qualification Gate v0.2

```

INPUT sample identity (r, c_t, A_i),
proposition package P_i, run_status
OUTPUT route_label, runtime status, covered
flag, failed_conditions
1 if run_status != COMPLETED: return (blank,
run_status, false, [RUN_STATUS])
2 failures <- []
3 for each required variable v in V_r:
4 if v is missing: append MISSING_v; continue
5 check C2-C9 against the frozen sample
identity and proposition record
6 check C10 for CAND-ASD-01-v0.2 or C11 for
every non-ASD Rule Card
7 route <- ELIGIBLE if failures is empty,
else NOT_ELIGIBLE
8 return (route, COMPLETED, true,
unique(failures))
9 scoring keeps ABSTAIN and ERROR as non-
coverage with blank route labels

```

A completed package is **ELIGIBLE** only when **failed_conditions** is empty. Otherwise Gate v0.2 returns **NOT_ELIGIBLE** and leaves the proposition records to show whether the problem was a violation or an unresolved requirement. **ABSTAIN** and **ERROR** have blank route labels. Neither confidence nor a downstream threshold can override these outcomes.

Plain language: Gate v0.2 **ADMIT** releases a completed package that passes every Rule Card check. Gate v0.2 **REJECT** withholds a completed package when at least one required check fails or remains unresolved.

3.6. Four frozen routes isolate proposition formation

All routes used the same case identity, assigned evidence, and error policy. B1 (B1-FROZEN-FIELD-v0.3-HO) searched the serialized text for a rule term and a number with a supported unit. It then returned a direct route and did not build

the shared proposition package. B1 is only a field-presence check. It does not test the configuration, endpoints, measurement meaning, source scope, conflicts, anchors, or other required conditions.

R0 (R0-DETERMINISTIC-RAW-v0.2-HO) used fixed expressions and rule-specific patterns before Gate v0.2. It read ranges and seat-width values, converted centimeters and inches to millimeters, checked target tokens, found competing values, and applied fixed seat-depth tests. If a value or condition was missing, R0 kept it as unresolved in a completed package. It did not search outside the assigned sources or add a pattern for an individual case.

L1 (L1-LUNA-PROP-v0.2-HO) called `gpt-5.6-luna` through the GFPLab Responses API with low reasoning and a 1,800-token output limit. Its frozen prompt supplied the Rule Card, exact target, serialized evidence, and typed JSON Schema. The prompt prohibited eligibility labels, retrieval, outside knowledge, target substitution, and unsupported completion. The runner retained the request, response, parsed output, and metadata, allowed one parsing attempt, and performed no repair. Provider-hosted inference may not be bitwise reproducible.

The text-only V1 prompt variant (V1-KAVLM-PROP-v0.1-HO) retained the frozen audit identifier. Its interface permitted only preassigned frozen visual assets, but none were assigned: all 88 formal records had an empty **visual_assets** field. The results characterize a text-only prompt variant and provide no evidence about visual input, OCR-free document understanding, or multimodal evidence qualification. V1 otherwise shared L1's Schema, parser, Gate, and runtime policy.

The registered Gold-proposition Gate served only as an interface upper bound. It was neither an automatic route nor a computed Holdout result.

3.7. Deterministic identity reconstruction remained a Development-only diagnostic

The v0.4 Development branch moved product identity out of normalized model fields. An evidence-only extractor received no target identity and returned at most three continuous source spans with evidence identifiers and anchors. Byte-level and character-level checks validated those spans before a frozen dictionary and parser reconstructed the identity.

A single reconstruction produced **MATCH** or **EXPLICIT_MISMATCH**. Missing, invalid, conflicting, or multiple reconstructions produced **UNRESOLVED**, which the wrapper mapped to **ABSTAIN** before Gate execution. Schema, parser, and interface failures continued to produce **ERROR**.

The v0.4 branch failed its prespecified Development-entry criterion and stopped before candidate freeze or blind evaluation. We report it only as a non-confirmatory diagnostic on this fixed Development set. The comparison records where identity authority was placed and how unresolved cases were handled; it does not establish causal improvement or better Holdout performance.

3.8. Hash-bound records preserve auditability and scope

Each formal run has its own saved record. It links the sealed case input and Rule Card with component versions, the request and response, parsed propositions, failed checks, route, runtime status, and seal; Gold is stored separately. A reviewer can retrace the run from those files. The hash chain establishes that the archived bytes are unchanged, not that the source was interpreted correctly.

CGFEV ends after it decides whether evidence may enter one rule execution. ELIGIBLE does not mean that the product passes the rule, and it is not a safety claim. Section 4 explains the Holdouts, Gold process, blind runs, denominator rules, and Development stopping rules used to test this boundary.

4. Experimental Design

4.1. Evaluation questions and evidence strata

The experiment evaluated evidence qualification before downstream rule execution. For each frozen evidence set E ,

engineering rule r , predeclared target product configuration c_t , and rule-configuration qualification contract K_{r,c_t} , the reference state was ELIGIBLE, NOT_ELIGIBLE, or DEFERRED. ELIGIBLE indicated that every contract condition required for authorization had been established. NOT_ELIGIBLE indicated an established disqualifying contract violation. DEFERRED indicated that the frozen evidence or qualification specification could not establish either reference outcome. The experiment did not evaluate whether the product subsequently satisfied the engineering threshold, and it did not treat document verification, evidence qualification, and product compliance as interchangeable claims.

Three questions governed the formal evaluation: whether Gold-ELIGIBLE evidence was released, whether Gold-NOT_ELIGIBLE evidence was admitted, and whether a route changed in the correct direction after one qualification condition was altered. Figure 3 and Table 3 keep the units for Holdout-A, Holdout-B, Cycle 3, and the identity audit distinct. Runtime and sealing details appear in Supplementary Table S3.

Table 3. Evaluation strata and analysis units

Stratum	Analysis unit	Frozen composition	Use and boundary
Development-33	Historical development sample	33 samples: 12-row frozen core + 21 historical rows; configuration audit: 13 EXACT_CONFIGURATION and 20 DEFERRED_DEFINITION entries.	Development/interface formation and retrospective diagnosis; not a formal test; never pooled with Holdout-A/B.
Cycle 3 Development	Rule-configuration instance; component call	18 instances across 4 deduplicated families; 40 first-pass component calls in a shared sealed batch.	Development-only method formation and entry check; 18 and 40 are distinct denominators; no generalization.
Holdout-A	Native record	48 frozen records: 47 binary (8 Gold-ELIGIBLE, 39 Gold-NOT_ELIGIBLE) + 1 Gold-DEFERRED; 17 official source files.	Family-isolated formal evaluation; DEFERRED remains visible but is excluded from the locked binary denominator.
Holdout-B	Member nested in controlled pair	40 members in 20 controlled pairs: 6 Gold-ELIGIBLE and 34 Gold-NOT_ELIGIBLE; raw annotator agreement 37/40.	Paired evaluation; members are dependent observations; pair endpoints are separate from member metrics.
v0.3/v0.4 audit	Same Development instance across versions	16 development-only cases: 8 ELIGIBLE, 4 configuration mismatch, and 4 DEFERRED.	Retrospective identity-binding and fail-closed audit; no blind test, no frozen candidate; DEVELOPMENT_ENTRY_NOT_PASSED.
Historical 13-case	Historical aggregate; unit unresolved	Complete Gold-output-scoring chain not located in the authoritative package.	Excluded GAP; no path, Gold, hash, scorer, or per-case result is reconstructed or quantified.

Note. The rows use different analysis units and are not pooled. We also omit the historical 13-case aggregate from every quantitative result table. Its full Gold-output-scoring chain is unavailable: no complete path, Gold record, hash, scorer, or case-level result could be reconstructed.

Evidence basis. Frozen Development audit; Holdout protocols and Gold artifacts; the Cycle 3 record; the Holdout-B annotation and adjudication record; and the PI-adjudicated v0.3/v0.4 audit decision.

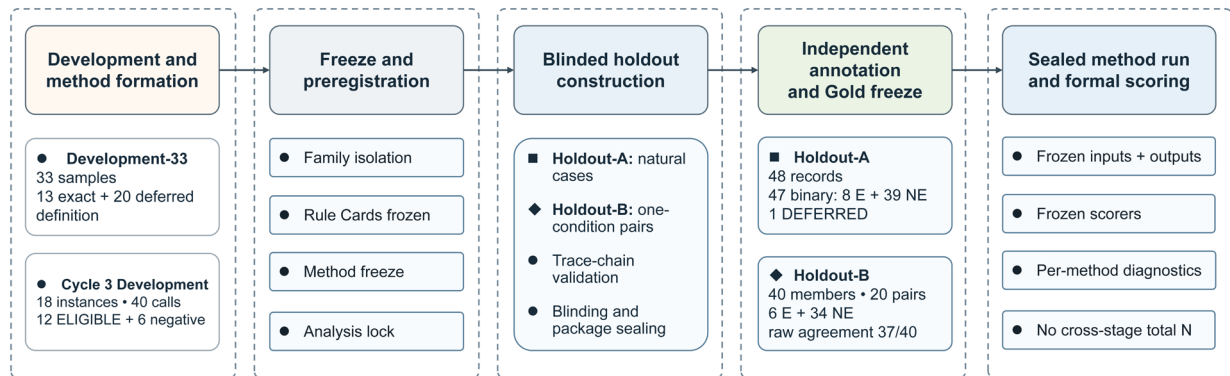


Figure 3. Evaluation protocol and evidence hierarchy

Figure 3 gives the order used in evaluation. We completed Development and froze the methods before constructing the family-isolated Holdouts. Gold was prepared by a separate process, and scoring was sealed later. E and NE in the figure mean Gold-ELIGIBLE and Gold-NOT_ELIGIBLE. Holdout-A is counted by record, whereas Holdout-B is counted by pair member. The 16-case identity study remains in Development. We did not use the old 13-case total because its case-level record cannot be reconstructed.

4.2. Holdout construction

Holdout-A was built from 17 official seating files whose hashes were frozen before replay. We first removed every family used in Development. The remaining files produced 48 natural cases, 12 for each Rule Card, and each route saw only the evidence assigned to its case. Records were kept even when a dimension was wrong, a version was disputed, or a poor text layer left only a visual locator; these were qualification conditions, not screening reasons. We selected cases without consulting trial-model failures or targeting a Gold balance. The construction package listed excluded Development families but did not include a standalone family-to-record table. The final blinded method input assigned a family ID to every case, and frozen Gold v0.5.1 retained the same mapping. The 48 cases represented 15 product families, with 2-6 cases per family.

Holdout-B contains 20 planned pairs (40 members). The source material was held constant within each pair except for one declared qualification condition. Depending on the pair, that change concerned identity, measurement meaning, conflict, stable-position support, a visual locator, or source scope. Pair analysis required Gold on both sides; a missing member was neither imputed nor replaced.

The two Holdouts contain no Development record. Cycle 3 uses 18 fixed Development cases for interface revision, and a separate 16-case set supports the identity comparison. Both sets document construction rather than independent evaluation.

4.3. Qualification contracts and Gold construction

Gold preparation was completed before method outputs were opened. Each case was tied to one Rule Card and an EXACT_CONFIGURATION target, and annotation followed the card's measurement object in millimeters. Seat depth, seat height, and seat-relative armrest height needed two endpoints. Adjustable seat depth also needed the mechanism and evidence of stable selectable positions; seat width needed one supported value or a source-stated interval. Overall dimensions, distance between arms, back width, and arm height from the floor were treated as different measurements.

The files and Gold labels were handled by different people. A freeze reviewer checked hashes, scope, configuration links, anchors, and conflicts without assigning a qualification state. Two annotators then worked independently from the fixed Rule Cards and without access to routes, one another's labels, acquisition strata, or prior decisions. Their sealed files were opened only for adjudication. Holdout-A does not retain enough pre-adjudication detail to reconstruct agreement [55]; the raw Holdout-B labels agree on 37/40 members.

4.4. Compared routes and proposition interface

The same case identity, evidence assignment, and error policy were given to B1, R0, L1, and V1. We fixed the route versions, prompt, Schema, target binding, and failure rule before a Holdout was opened. For diagnosis, the Gate was

also run on Gold propositions. That diagnostic is separate and is not treated as a fifth route.

For L1, one payload contained the Rule Card, exact target, serialized evidence, and Schema. We made one call to gpt-5.6-luna with low reasoning and a limit of 1,800 output tokens. The parser also had one attempt. Temperature and top-p kept the provider defaults. Search, retrieval, storage, and repair were off. The request, response, parsed result, and hash are kept in the sealed record.

L1 and the text-only V1 prompt variant used the same serializer, Schema, parser, and Gate. All 88 formal V1 records had an empty visual_assets field. The results therefore characterize a text-only prompt variant and provide no evidence about visual input, OCR-free document understanding, or multimodal evidence qualification.

4.5. Authorization gate and outcome semantics

Gate v0.2 checked the propositions against the target by using the tests in Section 3.5. Raw documents, Gold, model confidence, and the later engineering threshold were not available to it. Release required every test to pass. For a completed package, one failed test was enough to produce NOT_ELIGIBLE.

Gold retains three reference states, whereas a completed Gate v0.2 package receives one of two actions: ADMIT (ELIGIBLE) or REJECT (NOT_ELIGIBLE). REJECT withholds the evidence from this rule run; it is not a product-level violation. ABSTAIN and ERROR record runs that supplied no Gate action.

Holdout-A has one Gold-DEFERRED record, which is outside the 47 binary-Gold records. If a method admitted this case, a reviewer would still have to resolve it. One record is not enough to estimate DEFERRED recall. Holdout-B has only binary Gold labels.

4.6. Frozen runtime, blinding, and traceability

Section 3.4 describes the serializer and the order of its text windows. Supplementary Table S3 gives the provider settings, parser and serializer versions, and the setting used for the visual channel.

The wrapper parsed the output once and compared it with the sealed case. A malformed result, wrong evidence ID, or invalid anchor produced ERROR. We made no repair. Before the blind runs, we had fixed the inputs, targets, Rule Cards, prompts, Schema, Gate, serializer, parser, and runtime. Each route was run one time. The protocol did not allow search, evidence replacement, case-specific patches, threshold changes, or output repair.

Each run wrote to a new directory. It stored requests, responses, outputs, statuses, routes, logs, hashes, and the seal. Before Gold was opened, reviewers checked the sample set, method identity, hashes, logs, and blank route fields for ABSTAIN and ERROR. Scoring joined records by sample ID and stopped on a Schema, identity, state, denominator, or seal

mismatch. Any later correction needed a separate exploratory version; the primary output could not be overwritten.

HELM calls for explicit evaluation scenarios, metrics, and standardized conditions [56]. Jacovi et al. [57] recommend protecting evaluation data from test contamination. Kapoor and Narayanan [58] show how leakage and dependent train-test partitions can inflate results. Nosek et al. [59] recommend outcome-blind analysis plans and transparent reporting of deviations. CGFEV applies these principles through family-isolated Holdouts, one-shot frozen execution, sealed outputs, prespecified scoring boundaries, and auditable deviation records. The hash seals and stopping rules are study-specific controls, not procedures attributed verbatim to those sources.

4.7. Metrics and locked denominators

For Holdout-A, false admission used the 39 Gold-NOT_ELIGIBLE cases, while substantive decision coverage used the 47 cases with binary Gold. Gold-ELIGIBLE cases formed the denominator for eligible release. The original 40-case and 48-record summaries are retained in Supplementary Table S2.

$$\text{Eligible release rate} = \frac{N(\text{ADMIT} \wedge \text{Gold} = \text{ELIGIBLE})}{N(\text{Gold} = \text{ELIGIBLE})} \quad (11)$$

False admission counted authorization when Gold had established a disqualifying state:

$$\text{False-admission rate} = \frac{N(\text{ADMIT} \wedge \text{Gold} = \text{NOT_ELIGIBLE})}{N(\text{Gold} = \text{NOT_ELIGIBLE})} \quad (12)$$

Execution completion treated runtime failure as its own engineering endpoint:

$$\text{Execution completion rate} = 1 - \frac{N(\text{ERROR})}{N} \quad (13)$$

Substantive decision coverage required an ADMIT or REJECT, and therefore excluded ERROR and ABSTAIN:

$$\text{Substantive decision coverage} = 1 - \frac{N(\text{ABSTAIN})}{N - N(\text{ERROR})} \quad (14)$$

For a Gold-ELIGIBLE case, REJECT, ABSTAIN, and ERROR all prevented release. Nevertheless, route tables keep ADMIT, REJECT, ABSTAIN, and ERROR separate. We did not collapse false admission, eligible release, execution completion, and substantive decision coverage into a composite score.

Holdout-B was summarized at both member and pair level. Both-member correctness required two correct routes. For a Gold-discordant pair, flip-direction accuracy required the route to change in the ELIGIBLE/NOT_ELIGIBLE direction; pair coverage required substantive outputs for both members. These counts are descriptive. The initial

preregistration listed Wilson intervals [60], exact McNemar comparisons [61], and product-family cluster bootstrap as a sensitivity analysis [62]. Before Holdout execution, the accepted statistical lock specified product-family percentile bootstrap intervals for Holdout-A and exact sign-test intervals for Holdout-B. Supplementary Table S1 reports the locked Holdout-B direction intervals alongside post-scoring descriptive intervals for the other endpoints. The sparse or degenerate Gold-discordant pairs do not support confirmatory route comparisons.

Supplementary Table S1 reports the locked Holdout-B exact sign-test intervals for pair-direction correctness. Its remaining exact intervals are post-scoring Clopper-Pearson descriptive summaries; the table also reports the family-clustered bootstrap intervals specified by the final statistical lock. None of these intervals was used for route ranking.

4.8. Protocol deviations and post-unblinding amendment

Scorer v0.3 stopped after accessing the Gold structure because it expected samples[] rather than records[]. Candidate v0.4 recognized records[] but stopped on the DEFERRED encoding before entering the method-output loop.

DEFERRED already denoted unresolved reference evidence outside the binary Gold states. After accessing the Gold structure and before loading any sealed method output, we set the rule that excluded this record from the false-admission and coverage denominators. Authorization files, timestamps, stopped executions, control logs, and seals document the sequence, although no operating-system log independently lists every file read.

We then authorized Scorer v0.4.1, which produced the score. This post-unblinding scoring-protocol amendment was not fully prefrozen. It changed the false-admission denominator from 40 to 39 and the coverage denominator from 48 to 47. It changed no Gold label, input, method output, or route and triggered no rerun.

The primary result is B1 at 6/39. Supplementary Table S2 preserves the frozen-v0.3-compatible value of 7/40; R0, L1, and V1 remain at zero under both treatments.

Supplementary Table S2 shows the changes in false-admission and execution-completion denominators after excluding the Gold-DEFERRED record. B1 is the only route whose false-admission numerator changes, from 7 to 6; frozen route outputs, ERROR counts, and ABSTAIN counts remain unchanged.

4.9. Development-only audits and stopping rules

Cycle 3 follows 18 fixed Development cases through successive interface revisions. For each revision, we retained proposition and binding checks, route, release, false admission, ERROR, and severity. Because the sequence includes regressions, it is used only as a construction record and never as a Holdout estimate.

The identity audit compares v0.3 and v0.4 on another fixed set of 16 Development cases. Safe-correct denotes ELIGIBLE on an eligible case, NOT_ELIGIBLE on an explicit mismatch, or ABSTAIN on DEFERRED. Code assigns S2 to a local-field mistake, S3 to a major method error without false target admission, and S4 to target change or unsupported admission. In v0.4, code reconstructed identity from source spans; an unresolved reconstruction led to ABSTAIN. The branch missed its entry rule and stopped before candidate freeze or blind evaluation. We report it only as a Development audit. We report the counts as retrospective, non-confirmatory diagnostics.

5. Results

5.1. Formal evaluation differentiates qualification behavior

The datasets are kept separate in Table 3 and Figure 3. Holdout-A had 47 binary cases: 8 Gold-ELIGIBLE and 39 Gold-NOT_ELIGIBLE. Its one Gold-DEFERRED case is reported separately. Holdout-B had 20 pairs, or 40 members. Six members were Gold-ELIGIBLE and 34 were Gold-NOT_ELIGIBLE. The two annotators first agreed on 37 of 40 members. Development cases were not used in Holdout inference.

Table 4 and Figure 4 show different behavior across the Holdout-A routes. B1 released 4 of 8 Gold-ELIGIBLE cases and falsely admitted 6 of 39 Gold-NOT_ELIGIBLE cases. R0 completed all 47 binary cases and made no false admission, but it released 0 of 8 eligible cases. L1 and the text-only V1 prompt variant also had 0 of 39 false admissions and 0 of 8 eligible releases. L1 ended with 12 ERROR outcomes and V1 with 14. Their execution completion was therefore 74.5% and 70.2%. No route used ABSTAIN. Supplementary Tables S1 and S2 give uncertainty and denominator checks.

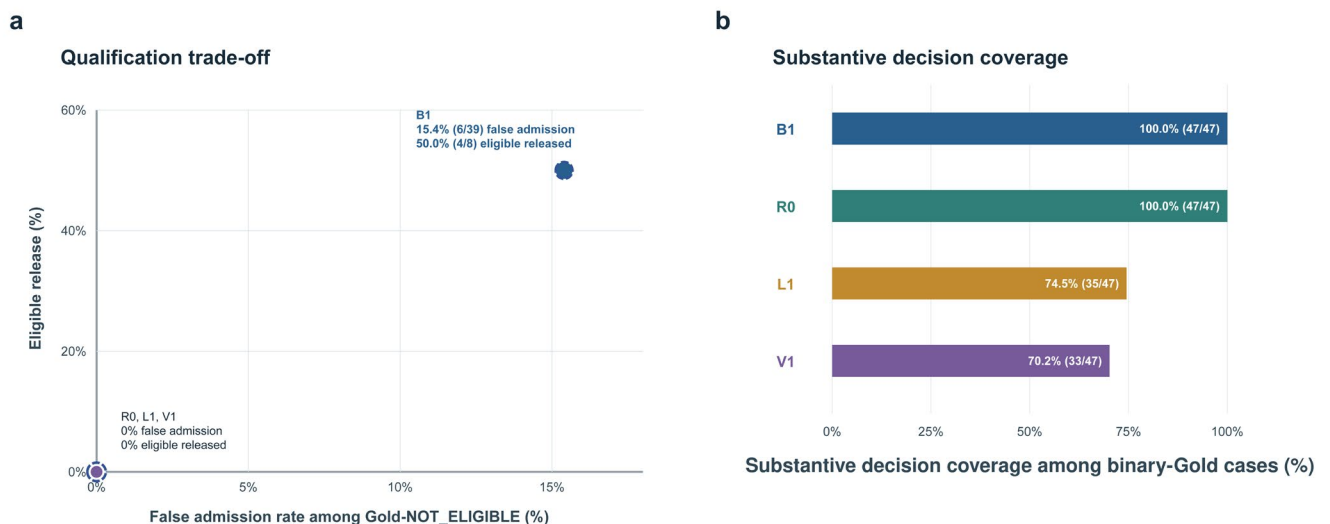
A zero false-admission count does not tell us whether eligible evidence was released. It also does not show whether the route completed. We therefore report eligible release, execution completion, and substantive decision coverage separately.

Table 4. Formal qualification results for Holdout-A and Holdout-B

Evaluation set	Method	Eligible release	False admissions	Execution completion	ERROR	Pair-direction correct	Both-end correct
Holdout-A	B1	50.0% (4/8)	15.4% (6/39)	100% (47/47)	0% (0/47)	N/A	N/A
Holdout-A	R0	0% (0/8)	0% (0/39)	100% (47/47)	0% (0/47)	N/A	N/A
Holdout-A	L1	0% (0/8)	0% (0/39)	74.5% (35/47)	25.5% (12/47)	N/A	N/A
Holdout-A	text-only V1 prompt variant	0% (0/8)	0% (0/39)	70.2% (33/47)	29.8% (14/47)	N/A	N/A
Holdout-B	B1	0% (0/6)	0% (0/34)	100% (40/40)	0% (0/40)	0% (0/4); covered 100% (4/4)	75.0% (15/20)
Holdout-B	R0	0% (0/6)	0% (0/34)	100% (40/40)	0% (0/40)	0% (0/4); covered 100% (4/4)	75.0% (15/20)
Holdout-B	L1	0% (0/6)	0% (0/34)	92.5% (37/40)	7.5% (3/40)	0% (0/4); covered 100% (4/4)	60.0% (12/20)
Holdout-B	text-only V1 prompt variant	0% (0/6)	0% (0/34)	77.5% (31/40)	22.5% (9/40)	0% (0/2); covered 50.0% (2/4)	50.0% (10/20)

Note. Counts are given with percentages. For Holdout-A, eligible release uses 8 Gold-ELIGIBLE cases, false admission uses 39 Gold-NOT_ELIGIBLE cases, and execution completion and substantive decision coverage use 47 binary-Gold cases. For Holdout-B, the corresponding denominators are 6 Gold-ELIGIBLE members, 34 Gold-NOT_ELIGIBLE members, and 40 members. Execution completion counts records without ERROR; substantive decision coverage counts records with neither ERROR nor ABSTAIN. The two values are equal here because no formal route abstained. The one Gold-DEFERRED Holdout-A record is outside the binary denominator. Both-member correctness uses all 20 pairs. Pair-direction correctness uses the four Gold-discordant pairs; V1's pair-direction denominator is 2 because it produced substantive outputs for both members of two such pairs. Supplementary Table S2 gives the v0.3-compatible 40-case and 48-record calculation. Exact uncertainty intervals are in Supplementary Table S1. Results are not pooled, and we do not claim an overall ranking or a best route.

Evidence basis. Sealed Holdout-A scoring v0.4.1; sealed Holdout-B primary scoring v0.2.1; and supplementary diagnostics recomputed after scoring from frozen Gold and sealed outputs. The historical 13-case aggregate is not used because its case-level Gold-output-scoring chain remains a documented GAP

**Figure 4.** Holdout-A qualification trade-off

Panel (a) reports eligible release for 8 cases and false admission for 39

Panel (b) reports substantive decision coverage over 47 binary-Gold cases

Execution completion is identical here because no formal route returned ABSTAIN

5.2. Route analysis isolates false admission

Table 5 and Figure S1 resolve the 39 Gold-NOT_ELIGIBLE cases by route. B1 admitted 6 cases, whereas R0 rejected all 39. L1 ended in ERROR 12 times and the text-only V1 prompt variant 14 times. None of the routes abstained.

The zero false-admission count therefore has two distinct explanations. R0 produced a substantive rejection for every

negative case. L1 and V1, in contrast, failed to complete 30.8% and 35.9% of those cases. Keeping the route counts separate avoids treating ERROR as a successful rejection.

Supplementary Figure S1 separates completed rejection from ERROR across the same 39 Gold-NOT_ELIGIBLE cases. This distinction shows why zero false admission has different operational meanings for R0, L1, and V1.

Table 5. Holdout-A negative-case routing

Method	Admit, n (row %)	Reject, n (row %)	Abstain, n (row %)	ERROR, n (row %)	Gold-NOT_ELIGIBLE total
B1	6 (15.4%)	33 (84.6%)	0 (0%)	0 (0%)	39
R0	0 (0%)	39 (100%)	0 (0%)	0 (0%)	39
L1	0 (0%)	27 (69.2%)	0 (0%)	12 (30.8%)	39
text-only V1 prompt variant	0 (0%)	25 (64.1%)	0 (0%)	14 (35.9%)	39

Note. Every row uses the same 39 Gold-NOT_ELIGIBLE cases. Admit is a substantive ELIGIBLE route and thus a false admission; Reject is a substantive NOT_ELIGIBLE route. Abstain records no substantive qualification decision. ERROR remains distinct and includes execution, Schema, parsing, or interface failure.

Evidence basis. The source for Table 5 was Holdout-A Gold v0.5.1 together with the four sealed JSON output files. We joined the records on sample_id. The totals after the join were checked against the frozen scoring summary.

5.3. Paired evaluation tests qualification-condition sensitivity

The paired analysis uses 40 members (Table 4). None of the routes released any of the six Gold-ELIGIBLE members. At the same time, false admission was 0/34 for the Gold-NOT_ELIGIBLE members. Four pairs had one Gold label on one side and a different label on the other. No route moved in the required direction for these four pairs. B1, R0, and L1 produced substantive routes for both members of each pair. V1 did so for two of the four pairs.

The records point to an upstream failure before a valid eligible baseline was formed. Across all four Gold-discordant pairs, B1, R0, and L1 returned NOT_ELIGIBLE on both sides; V1 also missed every eligible member and produced two positive-member interface or parsing errors. B1 found the rule term on the eligible members but missed the required numeric value-unit pattern. Proposition records from R0, L1, and V1 repeatedly left value, meaning, configuration, source scope, conflict, or rule-specific conditions unresolved. Future revisions should test exact-configuration binding, value-unit evidence formation, and Schema-parser alignment on these pairs. The records do not show that Gate v0.2 misclassified a valid eligible package; the independently evaluated pair members did not produce the expected route change.

B1 and R0 were correct on both members in 75.0% of the pairs. The value was 60.0% for L1 and 50.0% for V1. Most of this result comes from pairs whose two Gold labels were the same. When the labels differed, the eligible member was

still missed. We therefore keep pair correctness and direction as two results.

We repeated the calculation after excluding three disputed perturbed members. The sensitivity set contained 37 members and 17 complete pairs. Each route again released 0/6 eligible members and falsely admitted 0/31 ineligible members. Only one complete pair was Gold-discordant, and every route missed its eligible side. We left incomplete pairs out of the direction denominator.

5.4. Development revisions establish interface closure

Figure S2 shows what happened on the same 18 Development cases while the interface was being revised. L2.3 began at 3/18 correct, with 12 ERROR and 15 S3 events. L2.10 ended at 18/18; on the 12 applicable cases, its numeric, binding, and eligible-release checks were all 12/12, and the saved results contained no ERROR, S3, S4, or false admission among the six negatives. S3 denotes a major method error without false target admission; S4 denotes target/configuration change or unsupported admission.

The intermediate revisions were not monotonic. L2.6 reached 12/18 and recorded no ERROR, but six S3 events remained. L2.7 passed all 12 numeric and binding checks while end-to-end correctness fell to 9/18 and one ERROR reappeared. L2.9 reached 16/18 before L2.10 reached 18/18. L2.8 was a separate diagnostic and is not placed on this line.

Supplementary Figure S2 tracks the non-monotonic

Development trajectory from L2.3 to L2.10 and keeps the separate L2.8 diagnostic off the main line.

L2.10 passed every measured interface check on this fixed Development set. This is a method-building result, not an independent performance result. Qualification performance is taken from Holdout-A and Holdout-B.

5.5. Development audit of deterministic identity reconstruction and fail-closed routing

In v0.4, normalized model fields were replaced by source spans and locators followed by deterministic identity reconstruction (Table 6; Figure 5). Valid spans and locators were returned in 16/16 cases, but identity was reconstructed in 11/16. S3 identity decisions decreased from 2/16 under v0.3 to 0/16. Both architectures kept mismatch false admission at 0/4 and produced neither S4 nor ERROR; v0.4 routed two unresolved mismatches to ABSTAIN.

This stricter check failed closed more often. DEFERRED-to-ABSTAIN increased from 3/4 to 4/4, and total ABSTAIN increased from 3/16 to 8/16. At the same time, safe-correct routing decreased from 14/16 to 12/16, eligible release from 7/8 to 6/8, and explicit mismatch rejection from 4/4 to 2/4. Deterministic validation intercepted two local-field events in v0.4. The absence of the two previously observed hard identity decisions did not amount to complete identity recovery.

The v0.4 branch missed its prespecified Development-entry criterion and stopped before candidate freeze or blind evaluation. The observed counts suggest a descriptive trade-off between identity verification and substantive coverage on the Development set. They are non-confirmatory diagnostics on a fixed set and do not establish causal improvement, general error elimination, or better Holdout performance. The comparison uses the frozen v0.3.1 and v0.4 final_info JSON files and follows the PI adjudication.

Table 6. Development identity-binding and routing audit

Dimension	Metric	n / basis	v0.3	v0.4	Interpretive boundary
Identity	Identity tuple exact / identity reconstructed	16	87.5% (14/16)	68.8% (11/16)	Different version-specific identity measures; shown as the closest audit analogue.
Overall safe routing	Safe correct	16	87.5% (14/16)	75.0% (12/16)	Correct substantive route or prescribed fail-closed behavior under the audit definition.
Mismatch routing	Mismatch rejected	4	100% (4/4)	50.0% (2/4)	In v0.4, the two unrejected mismatches route to ABSTAIN, not ELIGIBLE.
Mismatch routing	False admissions	4	0% (0/4)	0% (0/4)	Mismatch false admission remains zero in the observed audit set.
Eligible release	Eligible released	8	87.5% (7/8)	75.0% (6/8)	Automatic release decreases under stricter reconstruction.
Deferred routing	DEFERRED to ABSTAIN	4	75.0% (3/4)	100% (4/4)	v0.4 sends every reference-DEFERRED case to ABSTAIN.
Substantive decision coverage	Total ABSTAIN	16	18.8% (3/16)	50.0% (8/16)	Fail-closed routing increases and substantive decision coverage decreases.
Error taxonomy	S2 events	16	0 (0%)	2 (12.5%)	Observed S2 events increase from 0 to 2.
Error taxonomy	S3 events	16	2 (12.5%)	0 (0%)	Observed identity hard-decision errors decrease from 2 to 0.
Error taxonomy	S4 events	16	0 (0%)	0 (0%)	No observed S4 event in either version.
Execution	ERROR events	16	0 (0%)	0 (0%)	No runtime/Schema/interface ERROR in either audit run.
Governance	Development entry passed	Fixed criteria	No	No	DEVELOPMENT_ENTRY_NOT_PASSED; no candidate frozen.

Note. Each percentage has its count. Both versions used the same 16 Development cases: 8 ELIGIBLE, 4 mismatch, and 4 DEFERRED. In v0.4, S3 fell to zero and mismatch false admission stayed at zero. But identity reconstruction, safe-correct routing, mismatch rejection, and eligible release also fell as ABSTAIN increased. The v0.4 branch missed its prespecified entry criterion and remained a retrospective, non-confirmatory Development diagnostic; it was never frozen or blind-tested. It cannot show better downstream LLM safety or rule-executor safety. The n / basis column gives the denominator or governance rule for each row.

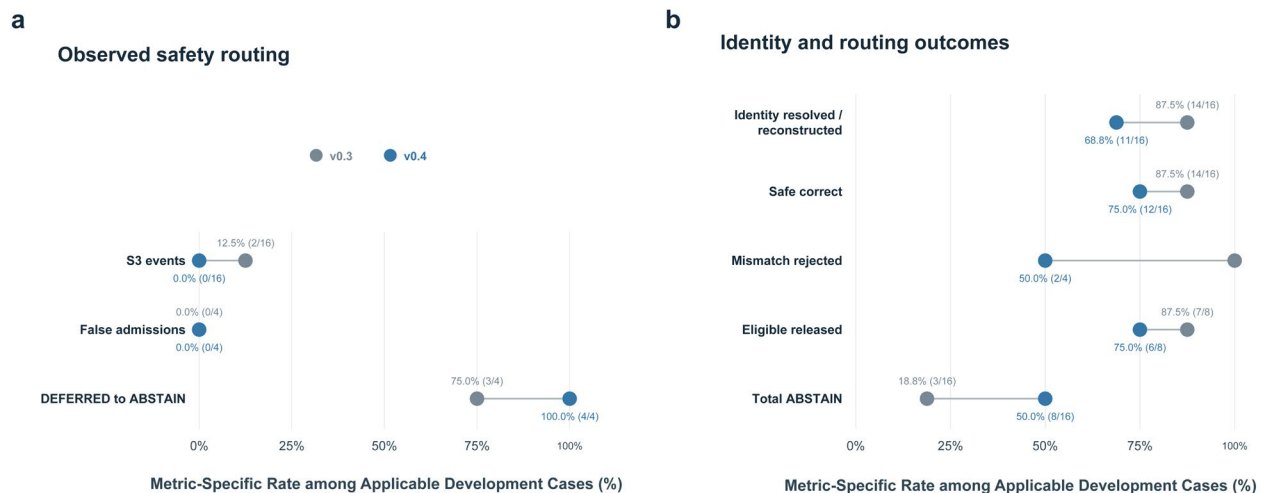


Figure 5. Development-only identity-binding diagnostic

Panel (a) reports S3, mismatch false admission, and DEFERRED-to-ABSTAIN

Panel (b) reports rebuilt identity, safe-correct routing, mismatch rejection, eligible release, and all ABSTAIN outcomes

Sixteen cases form the overall denominator. The denominators are four for mismatch, four for DEFERRED, and eight for ELIGIBLE. Identity was measured differently in the two versions, so those bars are used only as audit analogues. These retrospective Development bars do not support causal or Holdout claims.

5.6. Result synthesis and audit traceability

Once the audit package was frozen, two internal audit reviewers independently reopened the same evidence-output-score chain under no-search, no-rerun, no-rescoring, no-Gold-change, and no-contact attestations. They agreed on 111/112 case checks and all 13 metric checks. The PI adjudicated the remaining interpretive disagreements against the archived evidence and predefined Development-only claim boundaries, adopted qualified wording, and preserved the stricter dissent where the evidence did not support a broader claim. No reviewer workbook, Gold label, method output, score, metric, or denominator changed.

The trace also locates the outcome. B1 produced a route and released six disqualified cases. R0 usually reached Gate v0.2, which withheld the evidence. In 12 L1 cases and 14 V1 cases, execution stopped before Gate v0.2 returned a route. We kept these records as ERROR rather than recoding them as REJECT or absorbing them into an overall score.

This audit concerns admission only. It neither applies a downstream safety or compliance rule nor extends beyond the commercial-seating evidence used here.

6. Discussion and Conclusion

6.1. Evidence qualification creates an auditable engineering decision

For a completed case, a reviewer can open the assigned source, inspect the proposition and locator, read the Rule Card for the named configuration, and then examine the Gate record. Figures 1 and 2 show the same handoff. These items remain separate, so the proposal component never gains release authority.

When Gate v0.2 rejects a completed package, failed_conditions identifies the check that stopped it. ADMIT and REJECT are Gate actions; ABSTAIN and ERROR mean that execution supplied no action. In either situation, the record concerns use of an evidence package, not whether the chair meets the engineering requirement. Gold-DEFERRED remains a reference-annotation state; it does not create a production runtime queue.

6.2. Route-specific evaluation exposes qualification trade-offs

The four routes create different work for a reviewer. B1 released six disqualified Holdout-A cases. R0 returned a completed decision on all 47 binary cases but withheld all eight eligible ones. L1 and V1 left 12 and 14 cases, respectively, without a Gate action because execution ended in ERROR. Table 4 and Figure 4 report these events separately.

The paired cases reveal a second issue. A route can reject both sides and look correct when both Gold labels are NOT_ELIGIBLE, even if it ignores the condition that changed. Gold differed across four pairs, and no route changed in the required direction on any of them.

The diagnostic records support an upstream-failure

hypothesis. B1 failed to recover the required value-unit evidence, while term or value detection, proposition formation, or interface parsing prevented R0, L1, or V1 from forming a valid eligible package for Gate v0.2. The independently evaluated pair members therefore did not produce the expected route change.

Zero observed false admission therefore describes only one failure boundary. The tested routes did not jointly provide eligible release, completed execution, and response to a changed qualification condition. Exact-target binding was required by Gate v0.2; these results do not show that a route could recover that binding from the source while also meeting the other requirements.

6.3. Authority separation supports diagnosable improvement

On the 18 fixed Cycle 3 cases, the stored checks show where an observed failure appeared: interface, target binding, or final route. L2.10 recorded no remaining error on the measured checks for that set, although earlier revisions moved both forward and backward (Figure S2). We use the sequence to document method construction; independent qualification results come only from the Holdouts.

On the separate 16-case identity audit, v0.4 recorded two fewer S3 identity decisions than v0.3, while mismatch false admission remained 0/4 (Figure 5; Table 6). More identities were unresolved, and both safe-correct routing and substantive coverage were lower. We report these counts as a retrospective Development diagnostic; the branch was not frozen or blind-tested.

With the stored hashes, a reviewer can follow a case from its document and Rule Card to the response, propositions, route, score, and review decision (Figure S3). The chain supports record integrity. The reviewer must still

judge whether the document meaning was read correctly.

Supplementary Figure S3 links the frozen inputs, route files, Gold, scoring, denominator records, and later audit decision.

6.4. CGFEV structures manufacturing evidence handoffs

CGFEV comes after document extraction and before a compliance rule is executed. Research on configuration, extraction, provenance, selective prediction, and rule checking covers other parts of this path. We focus on one handoff: does this outside source support the named product, configuration, measurement, and rule conditions?

CGFEV differs from selective classification, RAG, attribute extraction, and schema constraints in retaining release authority for one rule-configuration contract at Gate v0.2.

We conducted a bounded check of current official web materials for two illustrative commercial seating families. Dauphin's mechanism page lists an 8 cm sliding-seat adjustment for the Indeed series and a 6 cm adjustment for the Indeed balance variant under Ball-Balance, a 20 mm within-family difference [63]. Sedus's current black dot page lists seat-depth adjustment of up to 60 mm, while its model-options area lists a 70 mm sliding-seat range, a 10 mm page-level difference [64].

Each example contained more than one official numerical value for the same broad dimension. Public manufacturer pages cannot support a defensible estimate of actual configuration-misattribution prevalence, and this check found no observed wrong-model assignment. The examples show why each value must remain bound to its product variant, mechanism, page section, and source state before rule execution. They remain separate from Holdout-A/B.

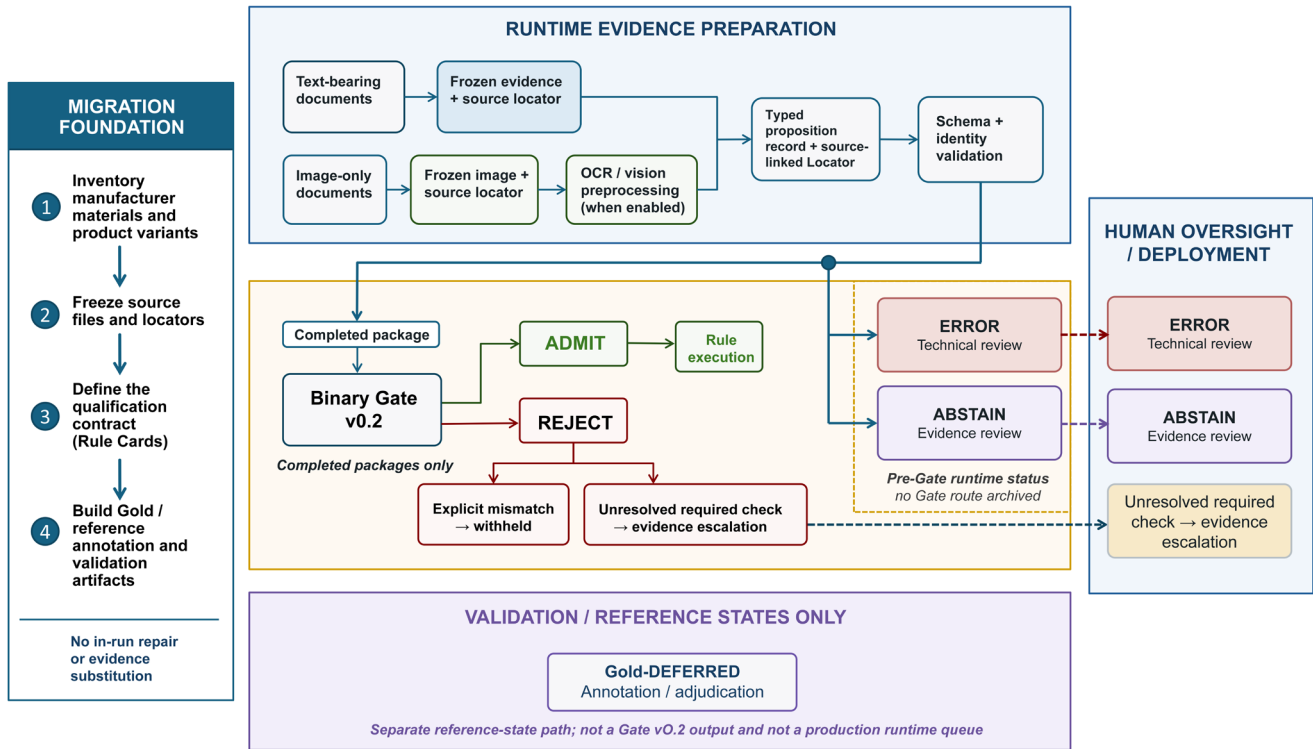


Figure 6. Implementation roadmap for moving unstructured supplier-document workflows into CGFEV. The roadmap separates frozen evidence preparation, pre-Gate runtime outcomes, executable Gate actions, deployment review, and validation-only reference states. Only completed packages reach Gate v0.2; Gold-DEFERRED does not enter the production runtime path.

Figure 6 translates these elements into a staged migration. Teams begin by inventorying manufacturer materials and product variants, freezing source files and locators, defining the qualification contract, and building Gold/reference validation artifacts. They should prioritize Rule Cards for dimensions with the highest engineering risk. Only a completed package reaches Gate v0.2; pre-Gate ERROR and ABSTAIN outcomes go to separate review queues. Gold-DEFERRED remains limited to validation annotation and adjudication. For image-only dimensions, optional OCR or vision preprocessing must preserve the original page region, derived text, and locator binding before proposition formation. The present V1 evaluation did not test this branch.

Adapting CGFEV to another manufacturing domain requires rebuilding the qualification contract. Domain experts must define product and revision identifiers, measurement meaning, admissible conditions, source hierarchy, conflict policy, and acceptable evidence anchors for each rule. The deterministic Gate can preserve the same release-authority boundary, but each domain requires its own Gold data and blind validation before transfer is claimed.

Prior traceability work records aircraft-part maintenance, modification, ownership, and movement across the part lifecycle [65]. Automotive E/E traceability binds product configuration and batch structure to process history, quality tests, resource data, and machine parameters [66]. Medical-

device traceability links requirements, specifications, design entities, tests, and manufacturing records [67]. These sources establish analogous record-binding needs, not CGFEV transfer performance. Each application would require domain-specific Rule Cards, document classes, approval roles, retention rules, and an independent validation set. The present seating results do not establish performance or safety effects in those domains.

6.5. Limitations and future validation

One protocol limit concerns the denominators. We gained technical access to the Gold Schema after the first denominator fields had been written, but before any route output was loaded. Two Holdout-A denominators were then amended under the procedure in Section 4.8. The amended values are primary, and Supplementary Table S2 preserves the earlier compatible calculation. Those fields were not fully frozen at the first stage.

Our interpretation is bounded by the frozen evidence. The study covers commercial seating, four rules, selected supplier documents, and two small Holdouts. Gold includes few eligible cases, and only four Holdout-B pairs cross the eligibility boundary (Table 4; Supplementary Table S1). All 88 formal V1 records had an empty visual_assets field. The results characterize the text-only V1 prompt variant and

provide no evidence about visual input, OCR-free document understanding, or multimodal evidence qualification.

The present implementation also leaves some distinctions unresolved. A completed NOT_ELIGIBLE route can reflect either an observed mismatch or an unresolved required check, while ERROR removes an L1 or V1 case before a substantive route exists. The wrapper verifies evidence assignment and anchor presence but does not remeasure each semantic field or compare every cited span character by character. The v0.4 identity branch missed its entry criterion and never entered candidate freeze or blind testing, and Holdout-B contains no prospective pair formed by adding a competing value.

Follow-up validation will initially target at least 60 instances from at least 15 previously unused product families, including at least 20 Gold-ELIGIBLE cases. It will use family-stratified sampling, separate Holdout-C primary and Holdout-D mechanism-challenge sets, genuinely bound visual assets, and prespecified eligible-release, false-admission, coverage, direction-switch, and human-triage endpoints. Before evaluation, the frozen protocol will set the final sample size from prespecified precision targets and define analyses that account for paired route outcomes and family-level dependence. Seat safety and final compliance remain separate downstream experiments.

6.6. Conclusions and outlook

We define a review point between a manufacturer file and one engineering-rule run. Rule Cards give the required conditions. Typed propositions keep the claim together with its location in the file. Code decides release, and hashes bind the saved record. In the evaluation, evidence release, false admission, and runtime failure are counted separately. The paired cases ask whether one stated change in the evidence leads to the required change in route.

The next problem is to pass more truly eligible evidence without adding false admissions. Further tests should use actual visual inputs and more product families. At this stage, our contribution is a defined qualification task, a clear split of decision authority, and a protocol for testing configuration-sensitive evidence.

Data availability

Summary data are included in the article and Supplementary Material. On reasonable request, the corresponding author will provide the frozen evaluation files, adjudicated Gold labels, Rule Cards, proposition Schema, and run manifests. Manufacturer documents remain subject to their copyright and access conditions.

Code availability

Requests for the frozen prompts, Gate code, scoring scripts, serializer specification, and runtime settings may be sent to the corresponding author. Because inference is provided through a hosted service, a later call may not reproduce the same bits exactly.

Appendix A. Supplementary Material

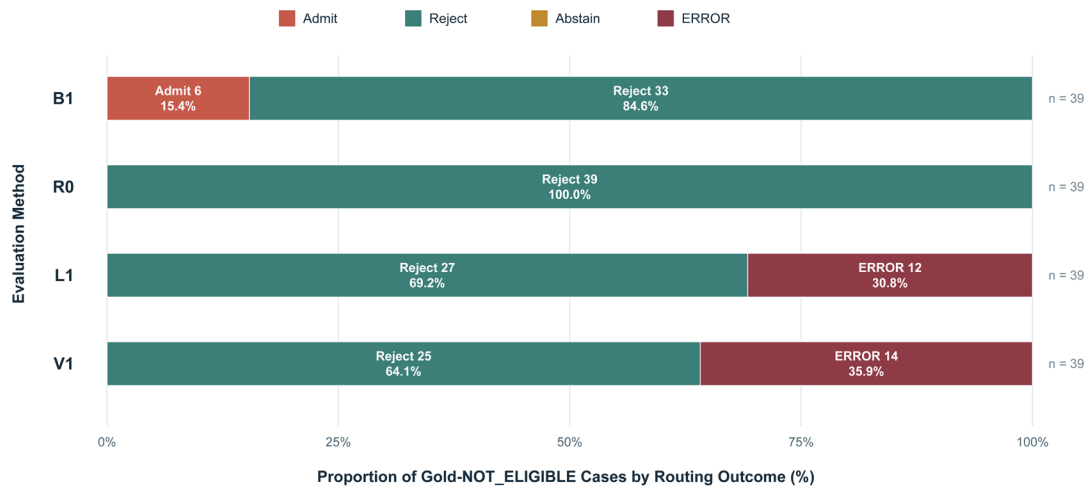


Figure S1. Holdout-A negative-case routing

All bars use the same 39 Gold-NOT_ELIGIBLE cases. REJECT is a completed NOT_ELIGIBLE route; ERROR means that no substantive route was returned.

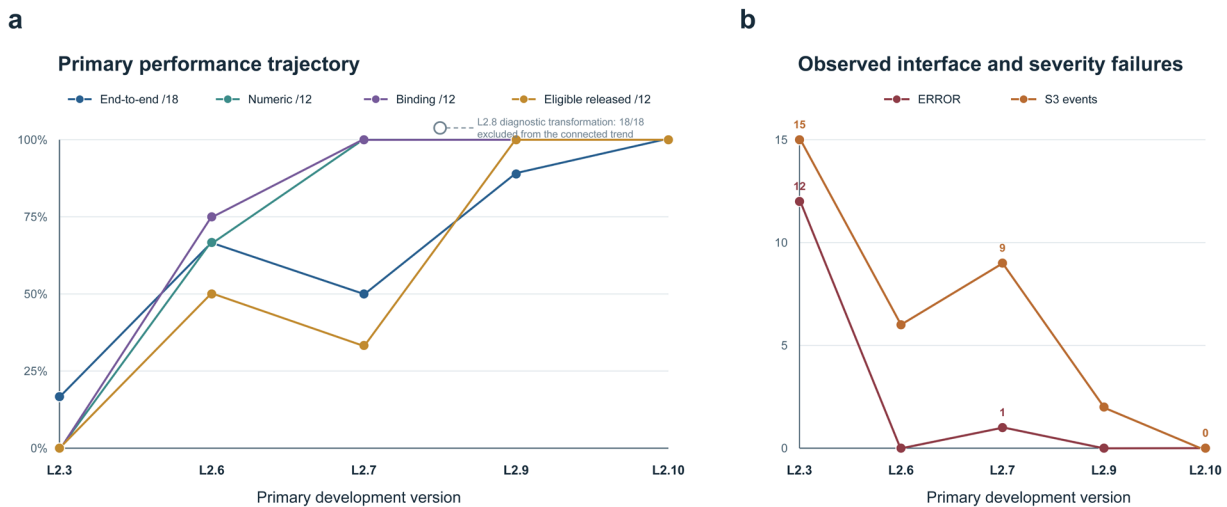


Figure S2. Development revision trajectory

L2.8 is excluded from the main line because it was a separate diagnostic. Panel (a) uses 18 cases for end-to-end correctness and 12 applicable cases for numeric, binding, and eligible-release checks; panel (b) records ERROR and S3. S3 denotes a major method error without false target admission, while S4 denotes target/configuration change or unsupported admission. No S4 event was recorded.

A quantitative claim is auditable only when every adjacent artifact transition is present and reviewable.

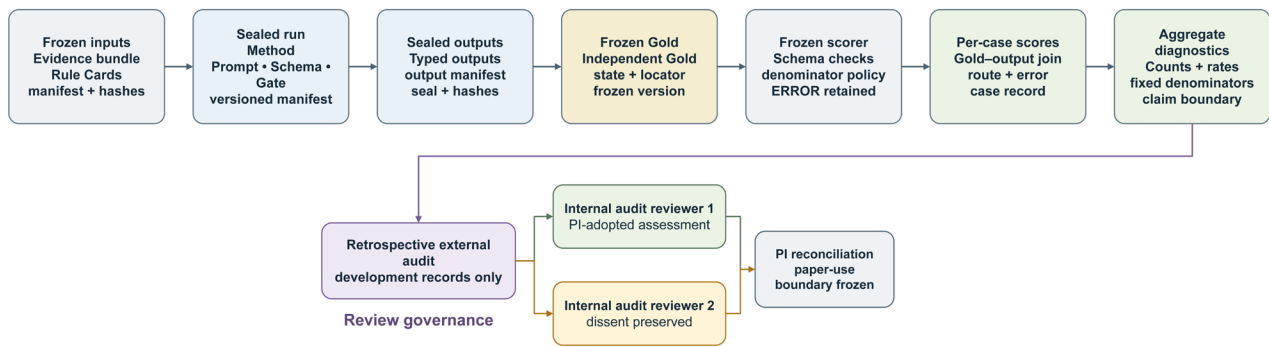


Figure S3. Frozen execution and audit chain

The audit followed frozen inputs through route files, Gold, scoring, and denominator records, while keeping Development audits separate. The PI adjudicated three interpretive disagreements against archived evidence and Development-only claim boundaries. No experimental artifact, Gold label, method output, score, metric, or denominator changed.

Table S1. Descriptive uncertainty for formal qualification endpoints

Set	Route	Eligible release, exact 95% CI	False admission, exact 95% CI	Execution completion, exact 95% CI	Pair direction, locked exact 95% CI	Both-end correct, exact 95% CI
A	B1	4/8; 50.0% [15.7, 84.3]	6/39; 15.4% [5.9, 30.5]	47/47; 100% [92.5, 100]	-	-
A	R0	0/8; 0% [0, 36.9]	0/39; 0% [0, 9.0]	47/47; 100% [92.5, 100]	-	-
A	L1	0/8; 0% [0, 36.9]	0/39; 0% [0, 9.0]	35/47; 74.5% [59.7, 86.1]	-	-
A	text-only V1 prompt variant	0/8; 0% [0, 36.9]	0/39; 0% [0, 9.0]	33/47; 70.2% [55.1, 82.7]	-	-
B	B1	0/6; 0% [0, 45.9]	0/34; 0% [0, 10.3]	40/40; 100% [91.2, 100]	0/4; 0% [0, 60.2]	15/20; 75.0% [50.9, 91.3]
B	R0	0/6; 0% [0, 45.9]	0/34; 0% [0, 10.3]	40/40; 100% [91.2, 100]	0/4; 0% [0, 60.2]	15/20; 75.0% [50.9, 91.3]
B	L1	0/6; 0% [0, 45.9]	0/34; 0% [0, 10.3]	37/40; 92.5% [79.6, 98.4]	0/4; 0% [0, 60.2]	12/20; 60.0% [36.1, 80.9]
B	text-only V1 prompt variant	0/6; 0% [0, 45.9]	0/34; 0% [0, 10.3]	31/40; 77.5% [61.5, 89.2]	0/2; 0% [0, 84.2]	10/20; 50.0% [27.2, 72.8]

Note. Pair-direction brackets are the two-sided 95% exact sign-test intervals specified by the final statistical lock; the scorer implemented them as Clopper-Pearson exact binomial intervals over direction-covered pairs. Pair-direction denominators include only Gold-discordant pairs with completed outputs for both members; V1 therefore uses 2 covered pairs. All other brackets are post-scoring two-sided 95% Clopper-Pearson intervals reported as descriptive summaries. No formal route returned ABSTAIN. Substantive coverage therefore has the same count as execution completion. These intervals describe this sample and were not used for route comparison. For Holdout-A, family-cluster bootstrap intervals for substantive coverage are [100, 100] for B1 and R0, [57.6, 89.8] for L1, and [53.2, 83.3] for the text-only V1 prompt variant. The sampling unit was the product-family identifier retained in the frozen blinded method input; all records from a sampled family were retained in each replicate. The mapping covered all 48 cases and contained 15 families. We used 10,000 replicates and seed 20260724. Evidence basis. Frozen Gold and sealed route outputs. Holdout-A supplementary diagnostics were recomputed independently of intermediate scorer rows, then reconciled with the sealed v0.4.1 summaries.

Table S2. Holdout-A denominator amendment sensitivity

Route	v0.3 false admission	v0.4.1 false admission	v0.3 execution completion	v0.4.1 execution completion
B1	7/40 (17.5%)	6/39 (15.4%)	48/48 (100%)	47/47 (100%)
R0	0/40 (0%)	0/39 (0%)	48/48 (100%)	47/47 (100%)
L1	0/40 (0%)	0/39 (0%)	36/48 (75.0%)	35/47 (74.5%)
text-only V1 prompt variant	0/40 (0%)	0/39 (0%)	34/48 (70.8%)	33/47 (70.2%)

Note. In v0.3, the non-admission denominator combined 39 Gold-NOT_ELIGIBLE cases and the single Gold-DEFERRED case; the primary v0.4.1 false-admission denominator contains only the 39 Gold-NOT_ELIGIBLE cases. Likewise, v0.3 used all 48 records for coverage, whereas v0.4.1 uses the 47 records with binary Gold. Gold, inputs, route outputs, ERROR counts, and ABSTAIN counts did not change. Evidence basis. Frozen v0.3 metric definitions, the sealed v0.4.1 score, the post-scoring governance review, and supplementary compatibility diagnostics.

Table S3. Experimental runtime and reproducibility specifications

Specification	Frozen technical setting	Runtime / operation	Boundary or interpretation
Rule Cards	Four cards: adjustable seat depth, seat height, usable seat width, and armrest height relative to seat; EXACT_CONFIGURATION; normalized unit mm.	Rule-specific required variables and necessary conditions.	The Gate authorizes evidence entry, not downstream product compliance.
Routes	B1-FROZEN-FIELD-v0.3-HO; R0-DETERMINISTIC-RAW-v0.2-HO; L1-LUNA-PROP-v0.2-HO; V1-KAVLM-PROP-v0.1-HO.	B1 field scan; R0 deterministic extraction; L1/V1 one proposition call per bound input followed by one parse/validation pass.	B1 is a shortcut baseline.
L1 / V1 model	gpt-5.6-luna; low reasoning; max_output_tokens=1800; store=false; external_search=false; repair_calls=0; provider temperature/top-p defaults retained.	OpenAI-compatible Responses API.	No hardware, operating-system, or undocumented provider specification is claimed.
Runtime envelope	CGFEV-HOLDOUT-RUNTIME-v0.2; Node.js major version >=24; frozen runtime declares no external Node packages.	API base/key supplied through GFPLAB_BASE_URL or OPENAI_BASE_URL and GFPLAB_API_KEY or OPENAI_API_KEY.	Only archived runtime fields are treated as experimental settings.
Evidence serialization	CGFEV-EVIDENCE-SERIALIZER-v0.1-HO; blocks sorted by stable evidence_id; preface contains Rule Card, target identity, and source metadata.	Short documents are supplied in full; long documents use deterministic windows in priority order: target/configuration, locator page, rule-keyword windows, then remaining pages; [NO_TEXT_AVAILABLE] is explicit.	Gold spans, coordinator comments, post-Gold OCR corrections, and manual transcription are prohibited.

Specification	Frozen technical setting	Runtime / operation	Boundary or interpretation
Visual channel	Only V1 may receive preassigned frozen page/image assets; page order follows evidence_id and page number.	Holdout-A/B visual_assets fields were empty.	The formal result is the text-only V1 prompt variant; no visual or multimodal effect is inferred.
Proposition interface	CGFEV-MINIMAL-PROPOSITION-SCHEMA-v0.1; one record per rule-required variable; range endpoints may be separate records; 13 required fields.	Remove Markdown fences; parse the first complete JSON object/array once; invalid fields, IDs, or anchors -> ERROR.	No imputation, repair, or second model call.
Gate v0.2	Checks C1-C12: completion, identity, variable completeness, mm units, semantics, exact binding, source/conflict, anchors, and ASD-specific stability/mechanism checks.	All checks pass -> ELIGIBLE; any failed condition -> NOT_ELIGIBLE.	ABSTAIN and ERROR retain blank route labels and are scored as non-coverage.
Sealing and blinding	Inputs, identities, Rule Cards, prompts, Schema, serializer/parser, Gate, runtime, and route versions frozen before blinded execution; Gold stored separately.	B1/R0/L1/V1 run once per bound input; create-new-only records include requests, responses, logs, hashes, and a manifest seal.	Sample-set equality, method identity, hashes, and status labels are checked before Gold unblinding; mismatch stops scoring.
Locked denominators	Holdout-A primary v0.4.1: 39 Gold-NOT_ELIGIBLE cases for false admission and 47 binary-Gold cases for coverage; Holdout-B: 40 members in 20 pairs.	v0.3-compatible 40/48 summaries are retained only in Supplementary Table S2.	No pooled accuracy is computed across Holdout-A, Holdout-B, Development, or the retrospective audit.

Note. These entries document frozen protocol settings and scoring boundaries, not further performance results. We infer no unrecorded hardware or operating-system build, undocumented provider parameter, or multimodal effect.

Evidence basis. Frozen runtime configuration and evidence-serialization specification, proposition Schema, deterministic Gate v0.2, sealed run manifests, and the authoritative evaluation protocol.

References

- [1] Sabin D, Weigel R. Product configuration frameworks: A survey. *IEEE Intelligent Systems*. 1998;13(4):42-49.
- [2] Aldanondo M, Vareilles E. Configuration for mass customization: How to extend product configuration towards requirements and process configuration. *Journal of Intelligent Manufacturing*. 2008;19(5):521-535.
- [3] Jiao J, Simpson TW, Siddique Z. Product family design and platform-based product development: A state-of-the-art review. *Journal of Intelligent Manufacturing*. 2007;18(1):5-29.
- [4] Simpson TW. Product platform design and customization: Status and promise. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*. 2004;18(1):3-20.
- [5] Opasjumruskit K, Schindler S, Peters D. Automatic data sheet information extraction for supporting model-based systems engineering. In: Luo Y, editor. *Cooperative Design, Visualization, and Engineering*. CDVE 2021. *Lecture Notes in Computer Science*. Vol 12983. Cham: Springer; 2021. p. 97-102.
- [6] Harper CA, Cox J, Kohler C, Scerri A, Daniel R Jr, Groth P. SemEval-2021 Task 8: MeasEval-Extracting counts and measurements and their related contexts. In: *Proceedings of the 15th International Workshop on Semantic Evaluation*. 2021. p. 306-316.
- [7] Göpfert J, Kuckertz P, Weinand JM, Kotzur L, Stolten D. Measurement extraction with natural language processing: A review. In: *Findings of the Association for Computational Linguistics: EMNLP 2022*. 2022. p. 2191-2215.
- [8] Rizvi STR, Mercier D, Agne S, Erkel S, Dengel A, Ahmed S. Ontology-based information extraction from technical documents. In: *Proceedings of the 10th International Conference on Agents and Artificial Intelligence*. Vol 2. 2018. p. 493-500.
- [9] Brinkmann A, Shraga R, Bizer C. ExtractGPT: Exploring the potential of large language models for product attribute value extraction. In: *Information Integration and Web Intelligence*. *Lecture Notes in Computer Science*. 2024. p. 38-52.
- [10] Raza M, Jahangir Z, Riaz MB, Saeed MJ, Sattar MA. Industrial applications of large language models. *Scientific Reports*. 2025;15:13755.
- [11] Xu Y, Li M, Cui L, Huang S, Wei F, Zhou M. LayoutLM: Pre-training of text and layout for document image understanding.

- In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2020. p. 1192-1200.
- [12] Kim G, Hong T, Yim M, et al. OCR-free document understanding transformer. In: Computer Vision - ECCV 2022. Lecture Notes in Computer Science. Vol 13688. Cham: Springer; 2022. p. 498-517.
- [13] Chen W, Wang H, Chen J, et al. TabFact: A large-scale dataset for table-based fact verification. In: Proceedings of the International Conference on Learning Representations. 2020.
- [14] Eastman C, Lee JM, Jeong YS, Lee JK. Automatic rule-based checking of building designs. *Automation in Construction*. 2009;18(8):1011-1033.
- [15] Ramonat M, Müller AW, Fay A. Towards automation of regulatory compliance checking in the product design phase. In: Proceedings of the 13th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management. 2021. p. 136-143.
- [16] Wu J, Zhang J, Debs L. Model validation for automated building code compliance checking. In: Construction Research Congress 2022. Reston (VA): American Society of Civil Engineers; 2022. p. 640-650.
- [17] Verhagen WJC, Bermell-Garcia P, van Dijk REC, Curran R. A critical review of knowledge-based engineering: An identification of research challenges. *Advanced Engineering Informatics*. 2012;26(1):5-15.
- [18] Zhang J, El-Gohary NM. Automated information transformation for automated regulatory compliance checking in construction. *Journal of Computing in Civil Engineering*. 2015;29(4):B4015001.
- [19] Solihin W, Eastman C. Classification of rules for automated BIM rule checking development. *Automation in Construction*. 2015;53:69-82.
- [20] Lewis P, Perez E, Piktus A, et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In: Advances in Neural Information Processing Systems. 2020;33. p. 9459-9474.
- [21] Thorne J, Vlachos A, Christodoulopoulos C, Mittal A. FEVER: A large-scale dataset for fact extraction and verification. In: Proceedings of NAACL-HLT. 2018. p. 809-819.
- [22] Barik AM, Hsu W, Lee ML. Time matters: An end-to-end solution for temporal claim verification. In: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track. 2024. p. 657-664.
- [23] Ge Z, Wu Y, Chin DWK, Lee RKW, Cao R. Resolving conflicting evidence in automated fact-checking: A study on retrieval-augmented LLMs. In: Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence. 2025. p. 9656-9664.
- [24] Es S, James J, Espinosa-Anke L, Schockaert S. RAGAs: Automated evaluation of retrieval augmented generation. In: Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations. 2024. p. 150-158.
- [25] Gao T, Yen H, Yu J, Chen D. Enabling large language models to generate text with citations. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. 2023. p. 6465-6488.
- [26] Huang L, Yu W, Ma W, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*. 2025;43(2):1-55.
- [27] Geifman Y, El-Yaniv R. Selective classification for deep neural networks. In: Advances in Neural Information Processing Systems. 2017;30. p. 4878-4887.
- [28] Geifman Y, El-Yaniv R. SelectiveNet: A deep neural network with an integrated reject option. In: Proceedings of Machine Learning Research. 2019;97. p. 2151-2159.
- [29] Guo C, Pleiss G, Sun Y, Weinberger KQ. On calibration of modern neural networks. In: Proceedings of Machine Learning Research. 2017;70. p. 1321-1330.
- [30] Ovadia Y, Fertig E, Ren J, et al. Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift. In: Advances in Neural Information Processing Systems. 2019;32. p. 13991-14002.
- [31] Brinkmann A, Baumann N, Bizer C. Using LLMs for the extraction and normalization of product attribute values. In: Advances in Databases and Information Systems. Lecture Notes in Computer Science. 2024. p. 217-230.
- [32] Geng S, Josifoski M, Peyrard M, West R. Grammar-constrained decoding for structured NLP tasks without finetuning. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. 2023. p. 10932-10952.
- [33] Norheim JJ, Rebentisch E. Structuring natural language requirements with large language models. In: 2024 IEEE 32nd International Requirements Engineering Conference Workshops (REW). 2024. p. 68-71.
- [34] Gebru T, Morgenstern J, Vecchione B, et al. Datasheets for datasets. *Communications of the ACM*. 2021;64(12):86-92.
- [35] Mitchell M, Wu S, Zaldivar A, et al. Model cards for model reporting. In: Proceedings of the Conference on Fairness, Accountability, and Transparency. 2019. p. 220-229.
- [36] Moreau L, Missier P, editors. PROV-DM: The PROV data model. W3C Recommendation. World Wide Web Consortium; 2013.
- [37] Zhang LL. Product configuration: A review of the state-of-the-art and future research. *International Journal of Production Research*. 2014;52(21):6381-6398.
- [38] Wang RY, Strong DM. Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*. 1996;12(4):5-33.
- [39] Strong DM, Lee YW, Wang RY. Data quality in context. *Communications of the ACM*. 1997;40(5):103-110.
- [40] Serra F, Peralta V, Marotta A, Marcel P. Use of context in data quality management: A systematic literature review. *Journal of Data and Information Quality*. 2024;16(3):Article 19.
- [41] Rebedea T, Dinu R, Sreedhar MN, Parisien C, Cohen J. NeMo Guardrails: A toolkit for controllable and safe LLM applications with programmable rails. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. 2023. p. 431-445.
- [42] Scholak T, Schucher N, Bahdanau D. PICARD: Parsing incrementally for constrained auto-regressive decoding from language models. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. 2021. p. 9895-9901.
- [43] Wu Y, Roesner F, Kohno T, Zhang N, Iqbal U. IsolateGPT: An execution isolation architecture for LLM-based agentic systems. In: Proceedings of the 2025 Network and Distributed System Security Symposium. 2025.
- [44] Nuyts E, Bonduel M, Verstraeten R. Comparative analysis of approaches for automated compliance checking of construction data. *Advanced Engineering Informatics*. 2024;60:102443.
- [45] Atanasova P, Simonsen JG, Lioma C, Augenstein I. Fact checking with insufficient evidence. *Transactions of the Association for Computational Linguistics*. 2022;10:746-763.
- [46] Arnold M, Bellamy RKE, Hind M, et al. FactSheets: Increasing trust in AI services through supplier's declarations

- of conformity. *IBM Journal of Research and Development*. 2019;63(4/5):6:1-6:13.
- [47] Cleland-Huang J, Berenbach B, Clark S, Settimi R, Romanova E. Best practices for automated traceability. *Computer*. 2007;40(6):27-35.
- [48] Zhao L, Alhoshan W, Ferrari A, et al. Natural language processing for requirements engineering: A systematic mapping study. *ACM Computing Surveys*. 2022;54(3):Article 55.
- [49] Mahmoud A, Niu N. On the role of semantics in automated requirements tracing. *Requirements Engineering*. 2015;20(3):281-300.
- [50] Ali R, Dalpiaz F, Giorgini P. Reasoning with contextual requirements: Detecting inconsistency and conflicts. *Information and Software Technology*. 2013;55(1):35-57.
- [51] Dianat I, Molenbroek J, Castellucci HI. A review of the methodology and applications of anthropometry in ergonomics and product design. *Ergonomics*. 2018;61(12):1696-1720.
- [52] Manavis A, Minaoglou P, Efklidis N, Kyratsis P. Digital customization for product design and manufacturing: A case study within the furniture industry. *Electronics*. 2024;13(13):2483.
- [53] Dalipi F, Kurti A, Ferati M. APPEND: A blockchain-based model of digital product passport for furniture industry. In: *Human Interaction and Emerging Technologies (IHET-AI 2024): Artificial Intelligence and Future Applications*. 2024;120. p. 95-103.
- [54] Schuitemaker R, Xu X. Product traceability in manufacturing: A technical review. *Procedia CIRP*. 2020;93:700-705.
- [55] Artstein R, Poesio M. Inter-coder agreement for computational linguistics. *Computational Linguistics*. 2008;34(4):555-596.
- [56] Liang P, Bommasani R, Lee T, et al. Holistic evaluation of language models. *Transactions on Machine Learning Research*. 2023.
- [57] Jacovi A, Caciularu A, Goldman O, Goldberg Y. Stop uploading test data in plain text: Practical strategies for mitigating data contamination by evaluation benchmarks. In: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. 2023. p. 5075-5084.
- [58] Kapoor S, Narayanan A. Leakage and the reproducibility crisis in machine-learning-based science. *Patterns*. 2023;4(9):100804.
- [59] Nosek BA, Ebersole CR, DeHaven AC, Mellor DT. The preregistration revolution. *Proceedings of the National Academy of Sciences*. 2018;115(11):2600-2606.
- [60] Wilson EB. Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*. 1927;22(158):209-212.
- [61] McNemar Q. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*. 1947;12(2):153-157.
- [62] Efron B. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*. 1979;7(1):1-26.
- [63] Dauphin HumanDesign Group GmbH & Co. KG. Dynamic sitting in the office: Ergonomic technologies by Dauphin [Internet]. Offenhausen (Germany): Dauphin HumanDesign Group GmbH & Co. KG; [cited 2026 Sep 3]. Available from: https://www.dauphin.de/dauphin/de/englisch/ergonomics/mec_hanisms
- [64] Sedus Stoll Aktiengesellschaft. black dot: Ergonomic swivel chair family [Internet]. Dogern (Germany): Sedus Stoll Aktiengesellschaft; [cited 2026 Sep 3]. Available from: <https://www.sedus.com/en/products/chairs/black-dot>
- [65] Ho GTS, Tang YM, Tsang KY, Tang V, Chau KY. A blockchain-based system to enhance aircraft parts traceability and trackability for inventory management. *Expert Systems with Applications*. 2021;179:115101.
- [66] Kuhn M, Franke J. Smart manufacturing traceability for automotive E/E systems enabled by event-driven microservice architecture. In: *2020 IEEE 11th International Conference on Mechanical and Intelligent Manufacturing Technologies*. 2020. p. 142-148.
- [67] Neelamkavil J, Pardasani A, Kernahan M. Traceability in medical devices design & manufacturing. In: *Proceedings of the Canadian Engineering Education Association (CEEAA)*. 2011.