Paid to measure, never to move the number

Direct score-contingent incentives and verifiable score construction in Watchdog.

Version 1.1, signed 2026-10-06 Â· revises version 1.0, signed 2026-09-02 Â· Canine Development

> **Changes in 1.1 (2026-10-06).** On 2026-09-26 Watchdog's pricing changed from volume-bucketed prices to four fixed-price plans, every one of which includes every feature; consumer reports are now sold through Assay, priced per engagement; and the reference scorer moved to the Code Assurance Initiative, which now publishes it as packages. The abstract, sections 3.1, 3.2, 3.4, 3.5, 4 (its figure and table), 5 and 10, and the sources are updated to match. Section 3.2 also now states that a product or portfolio report, which combines several reports, is charged after those reports are produced, not before the verdict as a single report is. The pricing model changed; the argument did not, and neither did any control it rests on.

Abstract

A software assessment is commercially conflicted the moment the assessor's pay moves with the result. A fee tied to a score, a band, a pass decision, a finding count, or a purchase the report triggers gives the assessor a stake in its own verdict, and no declaration of independence can dissolve that stake. Only the commercial model, the technical design, and evidence the recipient can inspect can do that.

This paper shows how Watchdog makes sure that what it is paid does not depend on the verdict it gives. Producers pay a fixed price for the plan they choose. Consumers pay a price for each report, and a single report is paid for before the verdict exists. Neither price depends on the score, the band, or the number of findings. Watchdog sells no remediation, bills nothing per finding, and puts no purchase instructions inside the signed evidence bundle. The reviewed revenue paths contain no identified score input, the reviewed score-aggregation boundary contains no identified customer or pricing input, and a build-time guard keeps both statements true on every commit, within the limits of a source scan.

The scoring itself is walled off from judgement. The open CAI reference scorer throws out everything marked advisory, including model-derived output, before the deterministic calculation. A recipient can recompute the score from the delivered evidence and rubric, verify the signed package, and compare the rubric-content digest it carries against the published catalogue.

The conclusion is strong but bounded. Watchdog is paid to do the assessment, not for the number it produces, and the parts that matter most can be checked rather than believed. The paper does not prove the absence of indirect commercial interests, the completeness of the upstream evidence, the repeatability of a fresh scan, or independent governance of future rubrics. Those limits are gathered in section 11, not hidden.

1. The question this paper answers

Every assessment system influences its own result. Its authors choose what to measure, how to classify evidence, how to weight dimensions, and where to place cutlines. That influence is unavoidable, so it is not the interesting question.

The interesting question is whether the assessor is paid according to the verdict.

This paper calls that **direct score contingency**: any commercial arrangement in which revenue depends on the score, the band, a change in score, a pass decision, the number or severity of findings, remediation generated by those findings, or a purchase the assessment triggers. Such an arrangement does not prove dishonesty. It creates a conflict that has to be recognised and governed.

Watchdog's claim is deliberately narrower than a claim of neutrality:

Watchdog is paid for conducting the assessment, not according to the verdict it produces.

The claim has two checkable parts. The pricing models take neither the score nor anything derived from it as an input. And the reviewed code contains no identified path by which commercial data reaches score aggregation, or score data reaches pricing.

The paper also covers the controls around that claim: rubric versioning and content digests, advisory exclusion, signed delivery, and score recomputation. They make parts of the system externally checkable. They do not rule out every indirect incentive, and the paper does not pretend they do.

2. Product scope, terminology, and how to read the claims

Watchdog analyses a codebase and its repository history, produces structured evidence, applies a versioned rubric, and delivers a signed report: a 0 to 100 headline CAI score, dimension-level results, bands, supporting evidence, and advisory material.

Term	Meaning in this paper
Producer	The organisation responsible for the assessed codebase.
Consumer	A party relying on the report: a customer, reviewer, auditor, governance function, or transaction counterparty.
Evidence bundle	The structured observations and dimension results supplied to the scorer.
Rubric	The versioned catalogue of rules used to aggregate eligible evidence.
Measured evidence	Evidence eligible to enter deterministic score aggregation.
Advisory evidence	Evidence marked as excluded from deterministic score aggregation.
Headline score	The 0 to 100 CAI result produced by the scoring calculation.
Headline band	The label assigned to the headline score.
Quality bar	A declared context that adjusts dimension-level band thresholds without changing the numeric score.
Recomputation	Re-running the score calculation against an existing evidence bundle.
Rescanning	Repeating the complete assessment from repository inspection through scoring.
Attested control	A control described and tested in a closed repository, not independently inspectable.
Public control	A control whose implementation or output can be inspected externally.

How to read the claims in this paper. Three phrases carry precise weight, and the paper relies on them rather than repeating their full meaning each time:

- **"No identified X"** means: searched for in the reviewed source at a specific revision, and not found. It is a finding, not a proof of absence.
- **"Attested"** means: checked by Canine Development in closed code. The reader cannot re-check it.
- **"Public"** means: the reader can check it without our help.

Four properties must also stay separate:

2.1 Commercial non-contingency

The verdict is not a direct input to any price calculation.

2.2 Aggregation separation

Commercial identity and pricing data are absent from the inspected score-aggregation boundary.

2.3 Reproducible calculation

The same evidence, rubric content, scorer implementation, and configuration produce the same score.

2.4 Measurement validity

The evidence faithfully and sufficiently represents the codebase.

This paper provides evidence for the first three properties. It does not, by itself, establish the fourth.

3. The commercial model

3.1 Producer pricing

A producer's bill is a function of the plan chosen, never of code quality. Watchdog is sold as four plans, each at one fixed price: Community free, Community unlimited and Professional, billed monthly, each with a fair-use allowance of line-scans a month; and Enterprise, a self-hosted licence billed yearly for a yearly line-scan allowance. Code volume decides which plan fits; the first survey of every repository is free on every plan. The only adjustments to a plan's price are coupons and referral credit, and they only ever reduce it.

We searched the reviewed pricing, billing, invoicing, subscription, payment, entitlement, coupon, referral, and usage-quota paths for a score, band, grade, or score movement: no identified reference. That review is attested and point-in-time; the build guard in section 5 holds the line between reviews. The public pricing page makes the same promise to customers: a fixed price for each plan, every price shown on the page, and no advertised score-contingent fee.

3.2 Consumer report pricing

Consumer reports are sold through Assay and priced per engagement: scoped per case by the number of targets, code volume and cadence, never by lines-of-code metering, and never from what the assessment finds. Where a report is charged inside the product, the charge is computed from what the code would cost to rebuild:

```
annual = multiplier × basis × rate + flat fee
perReport = annual / divisor
```

The basis is a cost-to-rebuild estimate from code volume, effort mix, and domain complexity. The multiplier is a persona coefficient; rate, flat fee, and divisor are separate commercial terms. The score and band appear nowhere in the formula.

Two repositories with the same rebuild estimate can still get different prices, because their other commercial variables differ. The claim is not "equal rebuild cost, equal price". The claim is that the verdict is not one of the variables.

A single report is quoted and paid before the verdict exists. A product or portfolio report, which combines the reports of several repositories, is not yet charged that way: each member report is produced first, and the charge follows. Its price comes from the same formula, so the verdict is still not an input; only the order in time differs, and the product is being changed so that these reports are also charged before the verdict. The timing of each charge is attested at the system level; for a specific engagement, the quotation, contract, invoice, and payment record are the proof, because a formula cannot show when money changed hands.

3.3 Watchdog does not sell the fix

Watchdog sells no remediation. Not per finding, not per fix, not by severity. When Watchdog finds a problem, Watchdog does not get paid to solve it.

This is the paper's central business-model distinction. Where an assessor is also paid for remediation arising from its own findings, every finding is potential revenue, and the assessment writes its own invoice. That pathway does not exist here. The point is structural, not an accusation: any vendor can adopt the same control, but adopting it means giving up the remediation revenue its own assessments generate, which is exactly why the distinction is meaningful.

3.4 No finding-linked module sale

Every plan includes every feature, so there is no module to buy in response to a finding. This principle has already cost us product copy.

An earlier runtime-readiness card, inside the signed evidence bundle, told the owner to enable a module: a purchase instruction wearing an evidence costume. We removed it. The card now reports the technical condition and sells nothing. That deletion is this paper acting as a design constraint rather than a description, and the absence has to be re-verified at each release, because a removed pathway can return.

3.5 Payment from producer and consumer

Watchdog can be paid by both sides of an assessment. The producer, whose code is assessed, pays for a Watchdog plan, and the consumer, who relies on the result, pays for the report. Neither of these prices depends on the verdict.

Two-sided payment is not neutrality; producer and consumer interests can align, reverse, or drift mid-engagement. What it creates is a countervailing constraint. Sustained leniency makes the report worthless to consumers, who need credible adverse evidence. Sustained severity makes it worthless to producers, who will stop buying an assessment they consider unfair. Lean either way for long and one of the two paying constituencies walks.

That is a commercial-structure argument, not something source inspection can prove. The code-verifiable claim stays narrower: both fees are independent of the score and band at the level of their direct inputs.

3.6 Credibility as economic exposure

What Watchdog actually sells is credibility: an assessment the producer can hand to a third party and the third party can trust. If the score were shown to be movable for a customer or in a preferred direction, the report would stop working as evidence, and both revenue lines would suffer at once.

The described model contains no identified direct revenue gain from moving the verdict, while exposed manipulation would destroy the asset being sold. That asymmetry is an economic reason to stay honest. It is not a substitute for verification, governance, or independent scrutiny.

4. Direct and indirect commercial pathways

Direct incentives link the result to revenue through a pricing or sales mechanism. In the reviewed model, every direct route from the verdict to revenue is barred, each by a named control:



Indirect incentives run through reputation, retention, positioning, and future relationships. They cannot be barred by a control, they do not all point the same way, and they deserve the nuance of prose:

Indirect incentive	Directional effect	Current position
Retention	Direction is indeterminate	High scores may reduce perceived need; low scores may motivate improvement or cause churn
General reputation and perceived accuracy	Favours credible calibration	Systematic leniency becomes uninformative; systematic severity becomes alarmist
Severity as a signal of rigour	Can favour harsher scoring	A plausible directional incentive, constrained but not removed
Transaction relationships	Depends on the engagement	A report's price does not depend on its result, and a single report is paid for before the result exists; future relationships may still matter
Broader product adoption	Direction is not fixed	Direct finding-to-purchase pathways are absent; indirect positioning effects remain
Rubric authorship	Can affect scores in either direction	Changes are versioned and digest-bound, but governance remains internal
Upstream evidence control	Can affect the delivered result	Evidence generation remains vendor-controlled and partly attested

4.1 Retention gives no direction to cheat in

A high score may please a producer, or convince them they no longer need us. A low score may prove our usefulness, or drive them away as unfair. Retention creates commercial pressure, but no stable direction to move scores in.

4.2 The market punishes both kinds of dishonesty

Score everyone highly and the assessment becomes uninformative; score everyone harshly and it gets discounted as alarmist. Market credibility pushes toward results that users regard as calibrated. That pressure is not proof of accuracy: a vendor can be sincerely wrong and stay credible for a while.

4.3 The one incentive with a direction: strictness sells

Being seen as demanding and hard to satisfy can be commercially valuable even when no formula reads the score. This is the clearest directional indirect incentive in the model. Producer acceptance constrains excess severity; consumer reliance constrains excess leniency. Neither constraint proves the chosen calibration is right, so this stays on the residuals list (section 11.4) as something to monitor, not a solved problem.

4.4 Pre-verdict pricing fixes today's fee, not tomorrow's relationship

Future reports, renewals, repeat business, referrals, partnerships, transaction relationships, and reputation all retain value that no pricing formula can account for. Those depend on the engagement. The paper claims the absence of identified direct verdict-linked revenue mechanisms, and nothing more.

5. Separation between commerce and score aggregation

The billing code has never heard of the score. The scoring code has never heard of the customer. That is the finding of the source review, in both directions: no identified score, band, grade, or CAI input in any reviewed revenue path, and no identified plan, subscription, entitlement, price, customer, or organisation input in the reviewed score-aggregation boundary. The crude corruption, a pricing service that reads the verdict or a scorer that knows who is paying, is not there.

A source review holds for one instant and can be broken by the next commit, by someone who never read it. So the review became a test. The build fails when a revenue path references a score, band, or grade, or when a scoring path references a customer, plan, price, or subscription. The guard names the paths it watches; rename or move one and a third assertion fails, so the isolation cannot quietly stop being enforced. A commit that makes price a function of the verdict, by the direct route, does not build.

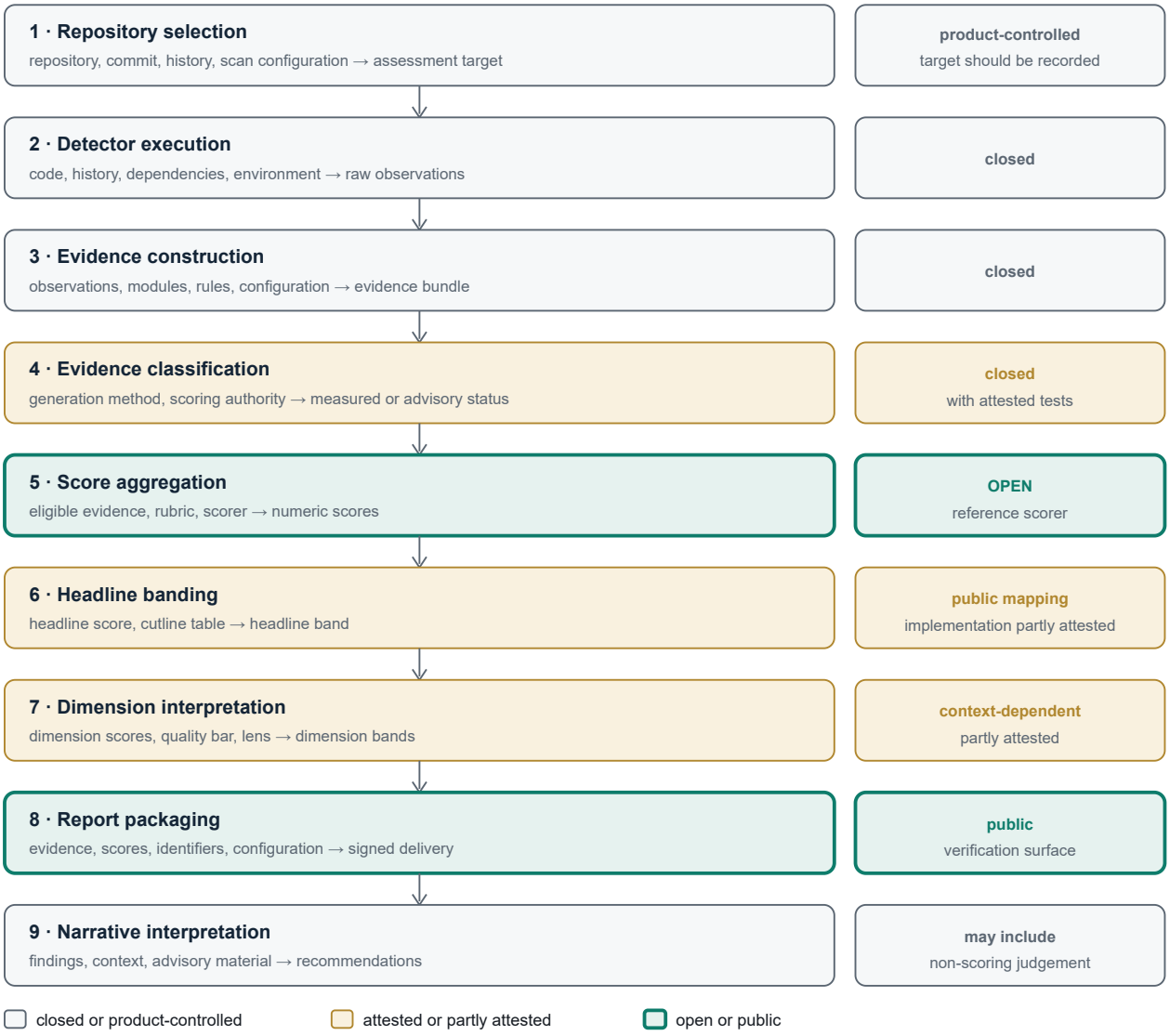
The defensible formulation is:

The reviewed implementation contains no identified direct dependency between the specified commercial variables and the inspected score-aggregation paths.

Now the limits, in one place. Commercial influence does not need the billing system to call the scorer; it can travel through rubric design, detector behaviour, scan configuration, module scope, evidence inclusion or suppression, quality-bar selection, release timing, defaults, or presentation, and none of those routes are touched by the guard. The guard also reads source, so generated code, reflection, configuration-mediated dependencies, and runtime data flow are outside what it can see. What it closes is the crude direct route, precisely and permanently; the subtler routes remain the substantive residual.

6. The complete measurement boundary

The scorer is open. It is also just one stage of nine.



The open reference scorer covers score aggregation. It does not independently verify repository selection, detector execution, evidence completeness or correctness, evidence classification, scan configuration, or contextual interpretation.

This boundary is central to the paper's credibility, because perfect arithmetic on incomplete evidence is still a wrong answer. Reproducible arithmetic is not measurement validity, and this paper never presents it as such.

For strong historical verification, a report should bind: repository and commit; history boundary; engine and detector versions; evidence schema version and bundle hash; enabled modules, scan profile and configuration; quality-bar selection; rubric identifier and content hash; scorer version; band-table and interpretation-table versions; signing key and time. The supplied material does not establish that every current report binds every item. The list is the target.

7. Rubric selection, versioning, and content binding

7.1 Shared version selection

The rubric lookup takes a rubric version, never a tenant or customer identifier: within that path, the rubric cannot be chosen by who is asking. The stronger statement, that no override exists anywhere in the product, remains attested, because the product repository is closed.

Rubric identifiers in a public paper should be either an actually released identifier or a generic format such as rubric-YYYY.MM.DD; a future-dated example must not be presented as an active release.

7.2 Version-name consistency

The public catalogue serves a rubric only when its declared rubricVersion matches the directory it is published in; a mismatch is withheld and surfaced. This checks naming. It does not detect an edited catalogue that keeps its version name.

7.3 Rubric-content digest

You cannot silently rewrite a rubric that someone holds a receipt for. That is what the content digest buys: each rubric version carries a canonical SHA-256 digest, published through the rubric registry, served by a per-version digest endpoint, and carried as rubricContentHash inside newly issued signed deliveries. A report holder can compare the digest in a retained package against the published catalogue at any later date.

This is detection, not prevention. We can still change a file we host; we cannot make that change silent to anyone holding an earlier signed package. The control is as strong as its parts: correct canonicalisation, correct deployment, the holder's retention of the package, and the availability of historical verification keys. The digest binds canonical content; it does not prove semantic equivalence in every implementation context.

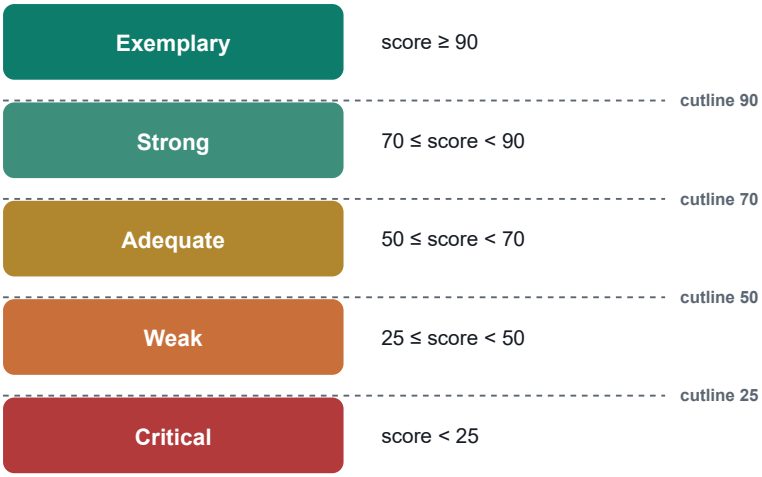
7.4 Future versions remain under company control

A new rubric version can change dimensions, weights, cutlines, and severity. Digests make the change visible; they do not make the decision independent or externally governed. Scores should be compared within a single rubric version unless a documented migration method exists.

Public change proposals, visible diffs, comment periods, and separately governed stewardship would strengthen legitimacy. They are not in place. Rubric governance remains a disclosed residual (section 11.3).

8. Score bands and contextual interpretation

The band is a label on the score, and labels can move without the number moving. The headline mapping is:



The intervals are half-open and cover the whole range. The mapping is absolute, not corpus-relative: no repository's band depends on how other customers scored.

Because a moved cutline changes the visible verdict without touching the number, the exact band table should be versioned or hashed and bound to the report.

Dimension-level bands additionally shift with a declared quality bar: Template/PoC, Preview, Production, or Mission-critical, with threshold offsets from minus 18 to plus 6 points, scaled by lens. The numeric score never moves; its label can. That is reasonable in itself, because a proof of concept and a mission-critical service deserve different expectations. It still matters to the trust argument, because recipients react to labels and colours, not just numbers. The report should therefore record the selected quality bar, who selected it and when, whether it could change after results were known, the offset table and lens scaling used, and the unadjusted numeric result.

The offsets are provisional and not yet empirically calibrated. That status stays explicit until a public calibration method exists.

9. Advisory output and deterministic scoring

9.1 Evidence classes

The advisory flag means one thing: this item never enters the deterministic calculation. It is broader than model use.

Evidence class	May contain an evaluative result	Model-derived	Enters deterministic aggregation
Measured evidence	Yes	Not under the described policy	Yes
Advisory model-derived evidence	Yes	Yes	No
Advisory descriptive evidence	Not necessarily	Not necessarily	No

The open scorer acts on the flag. It does not independently determine how the evidence was produced.

9.2 Layered regression coverage

Four layers guard the classification. The engine derives advisory status from its scoring authority rather than assigning it by hand; an invariant compares model-sampled classification against that authority for every registered dimension; a tally checks that the two classes cover the registered set; and the open reference scorer filters advisory dimensions and advisory meta-dimensions before aggregation.

The first three layers share one internal authority, so they are related checks, not independent proofs: if the authority misclassified a dimension, all three could agree and still be wrong. Only the fourth layer is independent at the point of enforcement. The honest description is layered regression coverage over one internal classification authority, combined with an independently inspectable public filter.

9.3 Negative-control test

A product-boundary test shows the flag doing the work. A bundle containing advisory items reproduces the claimed headline under the test assertion. Remove the advisory items and the score does not move. Flip one item from advisory to measured and it does.

That is strong regression evidence for the tested path, not a formal proof over every production path or delivered bundle. The publication candidate must state the equality rule the tests actually use, because "exactly equal" and "equal to six decimal places" are different claims.

9.4 The classification gap

A reader can inspect the rule that excludes advisory evidence. The engine logic that decides which items receive the mark is closed. Classification integrity is therefore attested, and this paper promises no timeline for changing that. The reader-checkable claim is the narrower one: the open scorer excludes whatever carries the mark.

10. Recomputation, rescanning, and verification

The deterministic relationship the open scorer establishes is:

```
same evidence bundle
+ same rubric content
+ same scorer implementation
+ same relevant configuration
= same recomputed score
```

It does not establish:

```
same repository commit
+ same rubric version
= same evidence bundle and same score on a new scan
```

A fresh scan additionally depends on detector versions, enabled modules, scan profile, environment, dependency availability, external services, missing-data policy, evidence schema and classification, and configuration. Scan repeatability is therefore a separate empirical property; recomputation does not establish it, and this paper does not claim it.

The public verification surface:

- https://codeassuranceindex.info/verify
- POST /api/verify
- POST /api/verify-delivery
- GET /api/registry/keys
- GET /api/rubrics
- GET /api/rubrics/{version}/digest
- GET /api/rubrics/{version}/catalog

Verification establishes that the package carries a recognised signature and is unchanged since signing, that it identifies a repository commit and rubric version, that the supplied evidence reproduces the claimed score under the supplied scorer and rubric, and, for a package carrying rubricContentHash, that it is bound to one canonical rubric digest.

Verification does not establish that the evidence is complete, the detectors accurate, the rubric appropriate, the scan repeatable, the system secure, or the assessment free of every commercial influence. It is not a certification.

The route is documented over HTTP because no anonymously installable package is yet published. The reference libraries, Cai.Scoring and Cai.Delivery, are published by the Code Assurance Initiative to GitHub Packages, which requires a signed-in GitHub account even to read; until an anonymous nuget.org feed exists, the reference implementation is built from source. The paper does not instruct readers to install a package they cannot fetch anonymously.

11. Residual limitations and governance questions

What this paper cannot show, gathered in one place.

11.1 Upstream evidence generation is closed

Detector execution, evidence construction, and most classification logic are proprietary. A recipient verifies the calculation on delivered evidence, not the process that generated it. Public detector specifications, conformance fixtures, dispute procedures, and external holdout evaluation would improve this without opening the engine.

11.2 Advisory marking is attested

The filtering rule is open; the assignment of the flag is not (section 9.4). The delivered classification is not independently verifiable, and no timeline is promised.

11.3 Rubric governance is internal

Canine Development chooses the dimensions, weights, cutlines, calibration methods, and release timing. Digests make changes detectable, not accountable to an external body.

11.4 Severity positioning remains a live indirect incentive

A reputation for strictness has commercial value (section 4.3). It should stay visible and be monitored through dispute rates, calibration studies, and customer behaviour.

11.5 The isolation guard is a source scan

The separation in section 5 is enforced on every build, but through source: generated code, reflection, configuration-mediated dependencies, and runtime data flow are outside what it can see, and it closes only the direct route.

11.6 Context moves the visible verdict

Quality bars shift dimension-level bands, and narrative interpretation includes advisory judgement. A stable number does not make every visible conclusion context-independent.

11.7 Report integrity is not measurement validity

A valid signature and a successful recomputation establish provenance and arithmetic. They do not establish truth, completeness, predictive validity, certification, security, or regulatory compliance.

11.8 Engagement records still matter

System-level pricing descriptions cannot prove the terms of one engagement. A recipient evaluating commercial independence should inspect the applicable quotation, contract, invoice, payment record, statement of work, and disclosures concerning other services.

12. Conclusion

A credible assessment makes the relationship between payment and verdict explicit, and then lets the reader check it. In Watchdog's model:

- The verdict is not a direct pricing input, on either side.
- Findings create no remediation revenue, because Watchdog does not sell the fix.
- The reviewed scorer boundary contains no identified commercial input, and the build guard keeps the direct route closed.
- The public scorer excludes advisory evidence, and that exclusion is inspectable.
- The signed delivery can be verified and the score recomputed from the supplied inputs.
- Rubric content can be pinned to a signed digest, so silent revision under a kept version name is detectable.

These properties are stronger than a promise of impartiality, because they attach to pricing rules, source boundaries, public code, signed artefacts, and repeatable calculation. They do not eliminate indirect interests: the engine is closed, the rubric is company-authored, scan repeatability is unmeasured, and severity still has positioning value. Section 11 keeps that list explicit.

The conclusion the evidence supports:

Watchdog is paid to do the assessment, not for the number it produces. The calculation can be checked, the rubric content can be pinned, and the discretion that remains is named rather than concealed.

Sources and evidence status

1. **Watchdog pricing page** https://watchdog.canine.dev/pricing Public evidence for the fixed-price plans and the absence of an advertised score-contingent fee.
2. **CAI reference repository** https://github.com/code-assurance-initiative/CodeAssuranceIndex Public implementation of the reference scorer, advisory filtering, rubric catalogue, and verification surfaces. The control record identifies CAI commit 1453fd5 for rubric-content digest support.
3. **Watchdog product pricing and scoring implementation** Closed and company-attested. Supports the producer plan pricing and the consumer pricing formula, the score/revenue source inspection and build guard, the headline band table, quality-bar behaviour, and the removal of purchase-shaped evidence copy. A scorer facade planned in this source would centralise score computation; it is not yet the enforcement point and is not used as evidence anywhere in this paper.
4. **Watchdog engine classification tests** Closed and company-attested. Supports advisory derivation, classification consistency, classification inventory, and the product-boundary negative-control test.
5. **Engagement artefacts** Quotation, contract, invoice, payment record, statement of work, and disclosures concerning related services. Engagement-specific evidence that compensation was agreed independently of the verdict.

Before publication, all attested implementation claims should be tied to the exact signing commits and re-verified against the production release.