

Applicability, Coverage, and Missing Evidence in Codebase Assurance Scoring

This paper explains how the current CAI implementation treats absence, partial coverage and producer-reported survey completeness. It does not create normative requirements. The current CAI specification, versioned rubric and reference scorer control in case of conflict.

Version	1.0
Date	17 September 2026
Signed	17 September 2026 by Jimmy Borch
Author	Jimmy Borch
Originating organisation	Canine Development
Series	CAI Founding Papers
Status	Informative, non-normative technical paper
Technical reference state	CAI repository through commit 1d557c36faa6f7650cfe6746db01665ad2ac9bfc

Contents

Abstract

1. Measured zero, omission and partial coverage

2. The current carriers of absence and partial evidence

3. Why omission can move the result

4. Partial coverage in the numeric fold

5. Applicability and survey completeness are reported separately

6. Reporting survey completeness with the headline

7. Artifact-level semantic boundaries

8. Four counterarguments

9. Empirical robustness under missingness

10. A reporting model for missing evidence

11. Limitations and maturity boundary

12. Conclusion

References

Verification record

Abstract

The Codebase Assurance Index (CAI) calculates a headline score from a producer-supplied evidence bundle. Measurements at dimension level are combined into categories; categories and direct meta-dimension inputs are combined into lenses; and the measured lens results enter an ordered weighted average (OWA) that produces the headline score and its reporting band. Evidence that is absent, partly covered or incomplete at different stages of this process has different arithmetic and interpretive consequences.

The current implementation distinguishes a measured failure from a missing measurement, a partially measured criterion from an absent criterion, and survey completeness from codebase condition. An omitted scored item receives no synthetic zero and is excluded from the set the fold consumes. A present dimension, the smallest scored unit, remains score-bearing when only partly covered: the scorer calculates `effective = score × coverage` before the reduced value enters the confidence-weighted mean for its category. Separately, the signed evidence can carry a producer-reported `surveyFit` object recording how many domain and architecture lenses applied to the scan and how many resolved. `surveyFit` is descriptive and non-scored; the reference scorer does not read it.[1][2][5]

Excluding an item from the fold does not make its absence arithmetically irrelevant. A category mean is recalculated over the contributors that remain. In the primary evidence path, a lens that produces no score drops out of the measured-lens set used by the headline OWA, and the positional weights are derived again over that shorter set. Omission can therefore move a score upward or downward even though the omitted item contributes no numeric value.[1]

The paper describes these semantics at the level recorded by the current implementation. The numeric evidence has no complete dimension-level taxonomy for the reasons an item may be missing, so assigning such reasons in CAI terminology would describe information the artifact does not carry. `surveyFit` provides higher-level visibility into applicable versus resolved survey depth without treating survey incompleteness as a code-quality penalty.[2][5]

The resulting model is asymmetric: absent evidence supplies no failure value, whereas weak partial evidence can reduce a score. The published rubric, scorer and tests make that behaviour inspectable. Its empirical robustness across a broad repository corpus remains a separate research question beyond the claims established in this paper.

1. Measured zero, omission and partial coverage

A composite score is usually read by people who did not perform the measurement. The report must therefore preserve the difference between a low result and a missing one. Missing evidence reaches a CAI result through two channels. The numeric channel is the fold, the fixed arithmetic that turns supplied measurements into a headline score; at that grain, a missing item is simply absent from the set being folded. The descriptive channel travels beside the number in the signed evidence and can state how much of the applicable survey resolved without entering the arithmetic. This section addresses the numeric channel, where the distinctions are narrow and their consequences can be shown directly.

The two channels answer different questions. The fold answers what result follows from the score-bearing evidence the producer supplied. Descriptive completeness records something about the reach of the survey that supplied it. A report loses essential context if it treats the first answer as proof that the second question has also been settled, but the arithmetic cannot repair that reporting error by itself.

A dimension is the smallest unit the fold consumes: one measurable property of the codebase, scored from 0 to 10. Its value is supplied by the producer, the party that performed the measurement and is accountable for it if the result is disputed. A deterministic dimension can contribute to the score, unlike advisory material, which accompanies the evidence for a reader but remains outside the number. The current implementation distinguishes three relevant states.

Case A: measured failure

```
score      = 0
coverage   = 1.0
confidence = 1.0
```

The dimension is present and score-bearing, so the fold uses its value. Coverage at 1.0 says the measurement reached the whole target surface for the criterion. Confidence at 1.0 says the producer applies no discount to the weight of this result relative to the other dimensions in its category. The value entering the calculation is the effective value, obtained by multiplying the score by coverage:

```
effective = 0 × 1.0 = 0
```

The resulting zero is evidence. It occupies the same place in the fold as any other measured result, with no special arithmetic treatment: it pulls its group down just as a 10 in the same position would pull it up. Its claim is narrow and explicit. The producer measured the criterion and supplied the lowest

score the scale permits. A reader may dispute that measurement, but the dispute concerns a stated result. The next two cases do not make the same claim.

Case B: no scored item

The dimension does not appear in the evidence bundle, the signed set of measurements the producer supplies for scoring. Nothing in the bundle names the dimension or marks it as skipped. The reference scorer performs the fold over the supplied items and creates no synthetic zero for an item the bundle never mentions.[1][2]

Several different events in the measurement process may produce that absence. An analyser may not support the relevant language, a check may never have been attempted, or the producer may have judged the criterion irrelevant to the codebase. These are different facts about the survey, but the current numeric artifact has no complete machine-readable dimension-level reason for them. Whatever happened upstream, the fold receives the same state: no scored item.

The artifact cannot support the interpretation:

this criterion failed.

Nor can it support, without other evidence, the interpretation:

this criterion was explicitly determined not applicable.

At the dimension grain used by the fold, the supported statement is narrower:

this item is absent from the scored evidence.

The producer may hold additional records that explain the absence, and a report may be able to cite them. Their possible existence does not add a reason to the CAI numeric artifact. This paper therefore distinguishes what may be known somewhere in the measurement process from what a recipient can derive from the signed fields and scorer behaviour under examination.

Case C: partial measured evidence

score = 8
coverage = 0.5
confidence = 1.0

The dimension is present, but the producer reports that the measurement reached half of the criterion's target surface. The scorer applies the same effective-value rule used in Case A:

$$\text{effective} = 8 \times 0.5 = 4$$

The effective result of 4 enters the category fold, where it is averaged with the other dimensions in the category and weighted by confidence.[1][2] The category receives the effective value rather than the raw score. From the category result alone, a reader cannot tell that this contributor covered half of its target surface; that fact remains in the dimension's coverage field. The item nevertheless participated in the fold, with a contribution reduced in direct proportion to the reported coverage. At coverage 0.5, it contributes half the value it would have contributed at full coverage, allowing a recipient who inspects the dimension to see both the measurement and its reach.

The distinction survives in the evidence even though it becomes less visible after aggregation. A category result of 40 could arise from fully covered low scores, partly covered higher scores or some mixture of the two. Preserving the dimension-level fields lets a recipient inspect which explanation applies instead of treating the category number as a complete description of the evidence behind it.

These three states are not interchangeable:

State	Meaning in the current numeric fold
Present score of zero	Measured score-bearing result at zero
Omitted dimension	No score-bearing item present
Present dimension with partial coverage	Score-bearing item with reduced effective value

The distinction preserves the trace from scored output to measured and signed evidence.

THREE STATES AT DIMENSION GRAIN

Measured failure score = 0 coverage = 1.0 confidence = 1.0 Score-bearing. Enters the category mean as 0. Effect: a measured zero	No scored item not in the bundle no coverage field no confidence field Outside the input set. No synthetic value is created. Effect: none directly	Partial measured evidence score = 8 coverage = 0.5 confidence = 1.0 Score-bearing at a reduced effective value of 4. Effect: reduced, not removed
---	---	--

The bundle records that an item is absent. It does not record why, and the fold receives the same state whatever the cause.

Figure 1. Three states at dimension grain. A measured zero is evidence, an omitted item is outside the input set, and a partly covered item is evidence at a reduced effective value.

A synthetic value inserted by the scorer would look identical to a value supplied by the producer while having no measurement behind it. Assigning zero to every missing item would therefore turn uncertainty or absence into a failure claim. It would also make the result depend on the reach of the producer's analyser: a criterion the analyser could not examine would score like a criterion the codebase genuinely failed.

Collapsing partial evidence into absence would create the opposite error. A weak partial measurement and a completely missing measurement would become indistinguishable even though the scorer treats them differently. The implementation instead preserves all three states: a measured zero contributes a number, an omitted item contributes none, and a partially covered item contributes a number already reduced by the coverage rule. Section 2 identifies the fields and absence patterns that carry these states; Sections 3 and 4 examine their consequences.

These differences are semantic as well as numeric. A measured zero is a claim about the code under a criterion. Partial coverage is a claim about both a measured result and the reach of that measurement. Omission, by itself, is only a statement about the contents of the scored bundle. Keeping the states distinct prevents the scorer from supplying claims that belong to the producer and prevents the report from presenting three different evidential conditions as one.

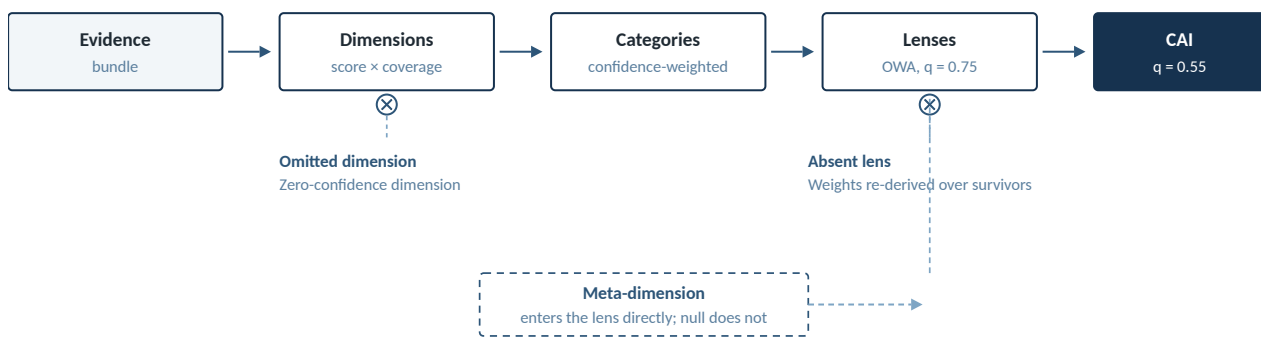
2. The current carriers of absence and partial evidence

The current implementation represents missingness and partial evidence through several fields and absence patterns rather than through one mechanism. Some remove an item from the arithmetic; coverage reduces an item's contribution while retaining it; and `surveyFit` remains outside the arithmetic, describing the survey rather than the code. The differences are visible in the scorer and must remain visible in reporting.

CAI performs the fold in stages. Dimensions first contribute to category scores, categories and direct meta-dimension inputs contribute to lenses, and measured lens scores contribute to the headline. A field that changes one stage may leave a different trace from a field acting at another. Describing all missingness as though it entered through one common gate would conceal where the set changed and what information remains available afterwards.

Current state	Numeric behaviour	What the current artifact can support
Dimension omitted	Does not participate in its category calculation	The scored item is absent; the numeric reason is not encoded
Dimension with confidence = 0	Adds zero category weight and is excluded from the critical gate, the band-capping rule for low contributors	Numerically unmeasured in the fold, although still present in the artifact
Dimension with coverage < 1	Uses <code>effective = score × coverage</code> ; remains score-bearing	Partial measured evidence
Meta-dimension score = null	Does not participate in the lens fold	No scored meta result
Lens absent in the primary evidence path	Does not participate in the headline ordered weighted average (OWA); weights are derived over measured lenses	No numeric lens result
<code>surveyFit.depthApplicable</code> / <code>depthFired</code>	Not read by the scorer	Producer-reported applicable versus resolved domain/architecture survey depth

Absence appears at three layers of the fold. Inside a category, a dimension may be omitted or carry zero confidence. At the input to a lens, a meta-dimension may be null. At the headline, an entire lens may produce no result. Although all three can be called absence, each interrupts different arithmetic. Coverage below 1 is the only state in the table that retains an item while reducing its contribution; `surveyFit` is the only one entirely outside the arithmetic. The following subsections examine these states from the dimension layer upward.



surveyFit travels beside the number: depthApplicable and depthFired, signed with the evidence.
The scorer does not read it. It cannot move the headline or the band.

Figure 2. Where absence enters the fold. Absence interrupts the arithmetic at three layers. Survey completeness travels beside the fold and never enters it.

2.1 Omitted dimensions

The evidence bundle carries the deterministic dimensions available for scoring.[2] Its contents state what the producer measured, and the scorer folds only those supplied values. An absent dimension is therefore outside the input set rather than a gap assigned a value. It contributes neither score nor confidence weight to the category mean: nothing enters the numerator or denominator on its behalf, so the mean is taken over a smaller set rather than diluted by a zero.[1]

When other dimensions remain, the category is calculated from those contributors, and the resulting category score contains no record that the set could have been larger. When no score-bearing contributors remain, the category has no numeric score and does not enter the lens fold.[1] Absence can thus propagate upward: a lens that loses all scored inputs reaches the headline layer as an absent lens rather than a low-scoring one.

The two outcomes differ in visibility. A category calculated from a reduced set still looks like an ordinary numeric category unless the underlying dimensions are inspected. A category with no numeric score can disappear from the lens input set altogether. The first form of absence changes the evidential basis while leaving a value; the second removes the value at the next layer. Both follow from the same omitted dimension mechanism.

The bundle has no dimension-level reason code for the absence, and the scored output cannot reconstruct one. A report can therefore state that a criterion went unmeasured but cannot derive why from the current numeric artifact. Every later interpretation in this paper is bounded by that limitation.

A future CAI version could decide to distinguish states such as:

- unsupported by the producer;
- not attempted;
- attempted but unresolved;
- explicitly not applicable;
- deliberately withheld.

The current numeric evidence provides none of this taxonomy. Each state, if it occurred, would reach the scorer in the same form: a dimension absent from the bundle. This paper therefore uses none of those labels as CAI dimension-level states. Section 5 describes the narrower applicability distinction available above the dimension layer.

2.2 Confidence zero

The evidence-model documentation states that a dimension measured at confidence zero should be omitted rather than represented as a zero-confidence measurement.[2] The convention reflects their equivalence in the arithmetic, but the current scorer accepts and processes a bundle containing a zero-confidence dimension. Omission is therefore a documented preference rather than a wire-level constraint.

Numerically, a zero-confidence deterministic dimension contributes nothing to the category numerator or confidence denominator: multiplication by zero removes both its effective value from the sum and its weight from the divisor. The critical gate, which allows one low contributor to cap the reported lens band, also excludes it because the gate considers only dimensions with confidence greater than zero.[1]

For the numeric category result, confidence zero is equivalent to omission. The paths differ in the bundle and converge in the arithmetic: one dimension was never supplied, while the other was supplied with no numeric weight. The equivalence ends at the calculation, however, because the artifacts can still differ structurally.

The zero-confidence dimension remains in the evidence bundle, and the current category result's `DimensionCount` is based on the number of grouped dimensions rather than only the positive-confidence subset.[1] A category holding one scored dimension and one zero-confidence dimension therefore reports a count of two while computing its mean from one. Treating `DimensionCount` as the amount of score-bearing evidence behind the category would overstate the evidence in this case.

This structural trace is why numeric equivalence must be stated narrowly. An auditor examining only the category score would see the same result after omission and after confidence was set to zero. An auditor examining the bundle and reported count can distinguish the two representations, even though neither path changes the weighted mean. The distinction matters whenever structural fields are used as evidence-volume indicators.

The careful statement is therefore:

Confidence zero is numerically equivalent to omission in the current category fold and critical-gate logic, but it is not necessarily identical in every reported structural field.

This formulation preserves the boundary between two source-level observations. The evidence-model documentation discourages confidence zero, while the arithmetic makes it equivalent to omission for the category fold and gate. Neither observation supports a claim of wire-level prohibition or artifact-wide identity.

2.3 Partial coverage

Coverage reduces an item's contribution without removing it. It is therefore the one field in the table through which the fold registers a measurement as incomplete rather than missing.

For a deterministic dimension:

```
effective = score × coverage
```

where coverage is a value from 0 to 1.[1][2] Like the score, coverage is supplied by the producer rather than derived by the scorer. It states how much of the criterion's target surface the measurement reached. Coverage is applied before the effective score enters the confidence-weighted category mean, so the category receives a reduced number rather than an unreduced value accompanied by a caveat.

Reduced coverage can therefore lower the category score even when the raw score is high. A dimension measured at 9 across one third of its target surface enters the mean at 3, below a dimension measured at 5 across all of its surface. The critical gate tests the same effective value, so coverage can also push a contributor below a threshold that its raw score would have cleared.

The current gate tests the effective contributor against the rubric's `criticalGate` threshold.[1][4] The rubric is the versioned catalogue that carries the scoring parameters as data, which is why the threshold belongs to the named rubric version rather than to the scorer. Under the current default scoring parameters, that threshold is `4.0` on the 0 to 10 contributor scale.[4]

For example:

```
raw score = 7
coverage  = 0.4
effective = 2.8
```

With positive confidence, this contributor remains in the category and falls below the current default critical threshold. The raw score of 7 would have cleared 4.0, but the producer-reported coverage reduces the effective value to 2.8. The gate receives that effective value without a separate distinction between a poor full-surface result and a strong raw result drawn from thin coverage.

That convergence is consequential for reporting. At the gate, a fully covered raw score of 2.8 and a raw score of 7 with coverage 0.4 are both low effective contributors. The dimension fields preserve how each effective value was formed, but the band cap itself records only that the threshold condition was met. A report explaining the cap must therefore retain the underlying score and coverage rather than infer the cause from the band.

Partial evidence thus remains in the fold at both points where it can affect the outcome: the category arithmetic and the critical gate. Coverage is applied once at the dimension, after which every downstream calculation uses the reduced effective value.

2.4 Null meta-dimensions

A meta-dimension is an input scored directly at lens level rather than at the level of an individual criterion, so it reaches the lens fold without first passing through a category mean.[1][2] A null-scored meta-dimension supplies no value and is excluded; the lens is calculated from the inputs that did produce values. An advisory meta-dimension is also excluded from the deterministic number, but for a different reason. It may carry a well-formed value and remains outside the fold because of its advisory classification rather than because a value is missing.

Null meta-dimensions place absence at the lens-input layer. The arithmetic effect has the same shape as dimension omission but a different reach: a missing dimension shrinks the set averaged within a category, whereas a null meta-dimension shrinks the set from which a lens is built.

2.5 Absent lenses

In the primary evidence path, the scorer receives a bundle of dimensions and meta-dimensions, calculates categories and lenses from the inputs that produce values, and then derives the headline.[1] A lens with no numeric score is excluded from the measured-lens set. This layer does not distinguish a lens with no applicable evidence from a lens whose evidence all went missing; neither appears among the inputs to the headline. The headline is consequently an aggregate of the measured lenses rather than of the full lens catalogue.

The OWA weights are derived over the surviving set and sum to one.[1] In this ordered weighted average, lens scores are sorted and weights attach to positions in the order rather than to named lenses, producing a worst-first aggregation. Renormalisation means recomputing the weights for the shorter ordered list; it does not transfer a departed lens's weight to a designated neighbour. Every surviving lens can therefore receive a different share.

The identity of a lens and its position in the ordered set are separate facts. When a lens disappears, the remaining lenses close the gap in the order and receive the weights associated with their new positions. The change therefore reaches beyond the missing lens: even a surviving lens with an unchanged score can exert a different influence on the headline after the set has shortened.

An implementation compatibility nuance applies to an older bundle shape. The scorer supports a thin fallback bundle containing pre-computed lens scores. In that path, positive supplied OWA weights that already sum to one within tolerance may be used directly; otherwise the scorer derives new across-lens weights.[1] The primary path always derives weights from the measured set, while the fallback may accept weights supplied by the producer.

The dimensions/meta-dimensions path is the founding CAI evidence path. The fallback is a compatibility route for supplying lens scores, not a separate missingness model, and it does not alter this paper's account of what happens when a lens is absent.

2.6 `surveyFit`

The signed evidence model can carry a `surveyFit` object.[2][5] Unlike the states described above, `surveyFit` remains outside the arithmetic and reports on the survey that produced the evidence rather than on the code described by that evidence.

Its principal fields are:

```
depthApplicable
depthFired
```

`depthApplicable` is the producer's count of domain and architecture lenses that applied to the codebase. Because not every lens in the catalogue is relevant to every codebase, the producer reports this count rather than the standard fixing it as a constant. `depthFired` counts those applicable lenses that actually resolved.[2][5] By construction it is a subset of `depthApplicable`, and the two values must be interpreted together.

Their difference is the producer-reported unresolved count at this level. It records how many surfaces the producer identified as applicable but did not resolve, describing survey reach without claiming what measurement would have found on those surfaces.

The object may also carry:

- `languageApplicability`;
- a producer-defined band;
- a producer-defined explanation.[2][5]

CAI carries these producer-supplied values inside the signed payload. Any alteration changes the signed content and is detectable, while the values remain attached to the result they describe. They are assertions by the party that ran the survey; CAI neither derives nor validates them. Integrity protection establishes what the producer sent, not whether the assertion was correct.

For reports exchanged between organisations, a valid signature can show that the completeness counts have not changed since the producer issued the bundle and can identify the party responsible for them. It cannot show that the producer classified every lens correctly or resolved every surface it says it resolved. Those questions require measurement evidence and review outside the signature check.

The current scorer neither calculates these fields nor folds them into the CAI result.[2][5] Given identical scored evidence, a bundle reporting that none of the applicable lenses resolved produces the same headline as one reporting that all resolved. Survey completeness and code condition remain separate propositions in separate fields, allowing a report to state both without using one as a proxy for the other. Coverage lies on the score-bearing side of this boundary because it describes the reach of an individual measurement rather than the survey as a whole. Sections 5 and 6 examine the reporting consequences.

3. Why omission can move the result

An absent item contributes no direct numeric value: the scorer adds nothing to a numerator, adds no weight to a denominator and inserts no synthetic figure. The aggregate is nevertheless recalculated from the values that remain. Because an aggregate depends on both its values and the set containing

them, removing an item can change the result even when the item contributes no replacement value. This is a general property of means and ordered weighted averages, not a CAI-specific anomaly. The following examples show the effect within a category and at the headline layer.

3.1 Category example

Take a synthetic category with two equally confident deterministic dimensions. The example is constructed to isolate one variable: both dimensions carry full confidence, so nothing in the weighting distinguishes them, and the only thing that changes between runs is which of the two is present.

```
A effective score = 2
B effective score = 8
confidence A      = 1
confidence B      = 1
```

The current category formula is:

```
Category =
  10 ×
    sum(effective × confidence)
  /
    sum(confidence)
```

Multiplying the confidence-weighted mean by ten places the category on a 0 to 100 scale while its dimensions remain on a 0 to 10 scale:

```
Category = 10 × (2 + 8) / 2
          = 50
```

If A is omitted, B retains the same score, coverage and confidence and becomes the only contributor:

```
Category = 10 × 8 / 1
          = 80
```

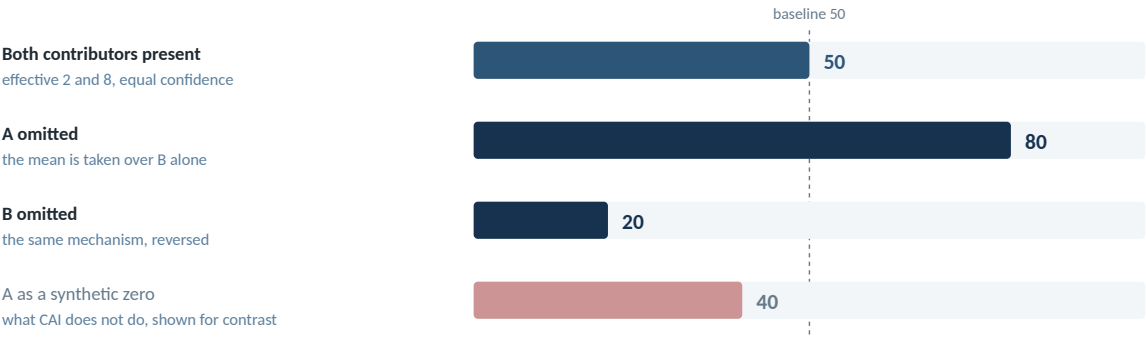
The result increases from 50 to 80 because the category is recomputed over the surviving contributor. The denominator falls from two to one, and B's 8 becomes the entire mean. If A had instead contributed a synthetic zero, the calculation would have been $10 \times (0 + 8) / 2 = 40$, producing a lower result rather than a higher one.

Now restore A and instead omit B:

```
Category = 10 × 2 / 1
          = 20
```

The result now decreases from 50 to 20. The mechanism is unchanged, but the direction reverses because the weaker contributor survives. An omitted contributor below the category mean raises that mean by leaving; one above the mean lowers it; and one exactly at the mean leaves the result unchanged. Omission is therefore neither a fixed penalty nor a fixed bonus. Its direction depends on a value the artifact does not contain: what the missing dimension would have scored. The result remains a well-formed mean, but it is a mean of a smaller set and answers a correspondingly narrower question.

OMISSION MOVES A CATEGORY MEAN



An omitted contributor below the mean raises it by leaving; one above it lowers it. A synthetic zero would lower the result instead of removing the item.

Figure 3. Omission moves a category mean. The direction depends on where the departing contributor sat relative to the mean. A synthetic zero would move the result the other way.

The category result alone cannot quantify the effect of the omission. Comparing the reduced bundle with a fuller earlier bundle may reveal that the category changed, but it still cannot establish what the missing item would have measured in the current run. The arithmetic demonstrates how a known removal changes a known example; it does not license a report to estimate an unknown missing value after the fact.

3.2 Lens example

The same set effect appears at the headline layer, where removing a lens can also change the positional weights. A category uses a confidence-weighted mean; the headline uses an ordered weighted average whose weights are derived for the number of measured lenses.

Consider three synthetic measured lens scores, chosen far enough apart that the rank weighting is visible:

40
70
90

The across-lens parameter q controls how quickly weight falls as scores improve. Each position in ascending score order receives q times the weight of the preceding, worse-scoring position; a value below 1 therefore produces worst-first aggregation. With the current default $q = 0.55$, the weights are approximately:

Ordered lens score	Weight
40	0.5398
70	0.2969
90	0.1633

The headline is approximately:

57.07

If the weakest lens, 40, is omitted, the remaining evidence stays unchanged. The two surviving scores, 70 and 90, receive new OWA weights of approximately 0.6452 and 0.3548 because the same q is distributed over a shorter ordered list. The headline becomes:

77.10

Omitting the 40 changes both the measured set and the rank weights, increasing the headline by approximately 20.03 points. Had the lens instead produced a measured score of 0, the ordered set would have been 0, 70 and 90, the original three weights would still apply, and the headline would be approximately 35.48. On these same three lenses, absence and a measured zero differ by about 41.62 points. The increase caused by omission reflects the removal of the lowest score from the aggregation, not a reward assigned to the missing lens.

Returning to the original set and omitting the strongest lens, 90, reverses the outcome. The remaining scores, 40 and 70, receive the same two-lens weights of 0.6452 and 0.3548 because the weights depend on the number and order of surviving lenses rather than their identities. The headline

becomes:

50.65

Omission now reduces the result by approximately 6.43 points. The runs differ only in which lens is missing, yet they move the headline in opposite directions and by different amounts. These deliberately synthetic calculations are derived from the published fold rather than sampled from a repository population. The scores were chosen to make the mechanism legible, and the arithmetic below permits direct reproduction. They establish directionality: headline omission can move the result either way, with the size of the movement depending on the departed lens's position. They provide no estimate of typical effects in real repositories; that requires the corpus study described in Section 9.

OMISSION MOVES THE HEADLINE

Three measured lenses



Weight attaches to position in the order, not to the lens.

Weakest lens absent



The same q is distributed over a shorter ordered list, so both survivors are reweighted.

Strongest lens absent



Identical weights to the row above: they follow the count and the order, not the identity.

Had the weakest lens instead produced a measured zero, the three original weights would still apply and the headline would be 35.48.
On these three lenses, absence and a measured zero differ by about 41.62 points.

Figure 4. Omission moves the headline. Ordered weights attach to positions rather than to lenses, so a shorter measured set is reweighted and the headline can move either way.

The asymmetry in effect size follows from worst-first weighting. The weakest lens initially occupies the position with the greatest weight. Removing it eliminates that low value, after which the weights are re-derived and the greatest weight attaches to the new lowest score. Removing the strongest lens instead eliminates a higher value from a less heavily weighted position. This explains the two synthetic results without turning either into a claim about how often comparable configurations occur in production evidence.

3.2.1 Calculation note for the synthetic OWA example

For this example, worst-first weights are derived from the current across-lens parameter $q = 0.55$ and normalised over the measured set:

$$w_i = q^{(i-1)} / \sum(q^j), \text{ for } j = 0 \dots n-1$$

For the ordered set 40, 70, 90:

```
raw weights      = 1, 0.55, 0.3025
normalised weights = 0.539811, 0.296896, 0.163293

headline =
  40 × 0.539811
+ 70 × 0.296896
+ 90 × 0.163293
= 57.071525
```

For 70, 90:

```
normalised weights = 0.645161, 0.354839
```

```
headline =  
  70 × 0.645161  
+ 90 × 0.354839  
= 77.096774
```

For 40, 70:

```
normalised weights = 0.645161, 0.354839
```

```
headline =  
  40 × 0.645161  
+ 70 × 0.354839  
= 50.645161
```

The paper therefore uses the rounded values 57.07, 77.10 and 50.65. The zero-scored counterfactual follows from applying the same three-lens weights to the ordered set 0, 70, 90. The arithmetic is explicit, but final publication should still verify the fixture against the pinned reference scorer identified in the technical reference state.

3.3 Interpretive consequences

When two reports use the same rubric but contain different measured sets, their headlines require evidence context. Each number may be a correct application of the named rubric to the supplied evidence, but the two are comparable only as well-formed aggregates over potentially different sets, not automatically as repeated measurements over identical ground. A recipient needs enough information to see which lenses produced scores and which categories fed them before judging whether the headlines support a direct comparison.

This is a scope issue rather than an arithmetic defect. The rubric and scorer can be applied correctly to each bundle while the resulting headlines describe different measured surfaces. Treating the shared scale as proof of shared evidential scope would add a claim that the fold never made. Conversely, exposing the measured set does not invalidate the headline; it tells the recipient what the valid calculation was a calculation of.

Lens-level results and, where available, survey completeness should therefore be exposed beside the headline. Both are already carried in the artifact, making their presentation a reporting decision rather than a change to the scoring standard.

Composite-indicator methodology supports the general relevance of this issue. The OECD/JRC handbook identifies missing-data treatment as a modelling choice that can affect a composite indicator, while JRC guidance notes effects on reliability and relative standing.[8][9] These sources do not define CAI semantics, and the handbook's imputation methods differ from CAI's decision to leave absent items without estimated values. CAI-specific behaviour comes from the scorer: every numeric claim in this section is derived from the published fold and can be recomputed from it. The external literature is used only to establish why missingness treatment requires explicit attention in a composite construction.

4. Partial coverage in the numeric fold

Coverage contains a measurement. Unlike the omissions examined in Section 3, a partly covered dimension has a value inside the arithmetic, reduced to reflect the reported reach of the measurement. Omission and partial coverage are therefore different states rather than two degrees on one missingness scale.

Suppose a producer reports:

```
score      = 8  
coverage   = 1.0  
confidence = 1.0
```

The effective score is 8 because full coverage leaves the raw value unchanged. If the same raw result is based on half the target surface:

```
score      = 8  
coverage   = 0.5  
confidence = 1.0
```

the effective score becomes 4. The raw judgement of the code is identical in both runs; only the producer's statement about the reach of the measurement has changed.

The scorer does not say:

We only saw half, so we will ignore this item.

It says, numerically:

This measured item contributes an effective value of 4 under the current coverage rule.

Applying the discount inside the fold is a substantive design choice. A system could instead retain the raw 8, carry coverage beside it as a caveat and leave any discount to the reader. CAI applies the rule consistently and automatically, although the resulting category number does not reveal on its own how much of the reduction came from coverage.

The rule also fixes where responsibility lies. The producer supplies both the raw score and coverage; the scorer applies their published relationship without deciding whether the stated coverage was appropriate. A recipient can recompute the effective value and challenge either input, but the fold introduces no discretionary adjustment. This makes the numeric treatment reproducible while leaving the validity of the underlying measurement open to review.

4.1 Coverage and confidence do different jobs

Coverage and confidence can both reduce a dimension's influence, and a category result alone does not reveal which field caused the reduction. The scorer nevertheless assigns them different roles at different points in the calculation.[1][2]

Coverage changes the value:

```
effective = score × coverage
```

Confidence changes the category weight:

```
Category =  
  10 ×  
  sum(effective × confidence)  
  /  
  sum(confidence)
```

A dimension with low coverage and positive confidence remains in the category with a reduced effective value. A dimension with confidence zero carries no numeric category weight, producing the arithmetic equivalence to omission described in Section 2.2. Coverage records how much of the criterion was examined; confidence records how much weight the producer assigns to the result. Their separation allows the artifact to represent both a thorough measurement with low confidence and a confident measurement of a limited surface.

Those cases should lead to different questions during review. Low confidence invites scrutiny of how much reliance the producer places on the result; low coverage invites scrutiny of which parts of the target surface were reached. Two dimensions may exert the same eventual influence on a category through different combinations of score, coverage and confidence, yet the evidence fields preserve the different reasons. A report that shows only the category result discards that diagnostic value.

4.2 Coverage can interact with the critical gate

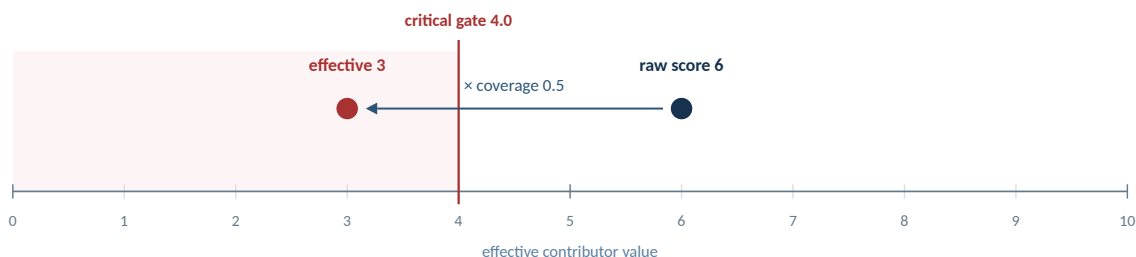
The critical gate evaluates the effective contributor after coverage has been applied.[1][4] Coverage can therefore move a dimension below the gate threshold even when its raw score would be above it. At the point of evaluation, the gate receives the effective number rather than a separate raw-score and coverage explanation.

Example:

```
raw score = 6  
coverage  = 0.5  
effective = 3
```

Under the current default critical threshold of 4.0, a positive-confidence, non-advisory contributor with an effective value of 3 can cap the displayed lens band.[1][4] A band is the categorical label assigned through rubric-defined cutlines. The gate changes that label without changing the numeric lens score, allowing one weak effective contributor to limit how an otherwise higher average is classified.

The number and band therefore carry related but non-identical information. The numeric lens score remains the aggregate produced from the lens inputs, while the band adds a rubric-defined interpretation, including the possibility that one critical contributor prevents a higher label. The gate itself changes the band rather than the numeric lens score. When coverage moves a contributor below the threshold, the underlying dimension fields explain why the cap occurred.



Coverage is applied once, at the dimension. The gate tests the effective value, so a strong raw score measured over a thin surface can cap a lens band that its raw score alone would have cleared. The cap records only that the threshold was met; the dimension fields record how the effective value was formed.

Figure 5. Coverage and the critical gate. Coverage is applied once at the dimension, and the gate tests the reduced effective value rather than the raw score.

Partial coverage can therefore affect both:

- category arithmetic through the reduced effective value;
- lens interpretation through the critical-contributor gate.

An omitted dimension follows neither route. It changes its category by leaving the set, as Section 3 showed, and never reaches a gate that can test only present contributors.

4.3 The asymmetry

Taken together, Sections 3 and 4 establish an asymmetry in the current system:

Partially measured weak evidence can reduce the score, while fully absent evidence does not participate.

A criterion that is measured thinly can reduce the score, whereas an unreported criterion cannot reduce it directly. This is established, testable behaviour in the published fold. Classifying the asymmetry as a defect would require a normative position on how partial and absent evidence ought to affect a score, which is a separate standard-design question. The same evidence also cannot establish that the policy is empirically robust across real repositories. That claim would require a frozen corpus and the perturbation protocol outlined in Section 9. The present paper can describe the rule accurately while leaving both its normative defence and empirical optimality open.

Three claims must remain separate. First, the implementation behaves asymmetrically; that is established by code, tests and arithmetic. Second, the asymmetry is justified as a policy; that requires a normative argument about what CAI should reward or penalise. Third, the policy produces stable and decision-useful results across repositories; that requires empirical evidence. Conflating these claims would either understate the implementation fact or overstate the support available for the design.

5. Applicability and survey completeness are reported separately

At dimension grain, the numeric fold represents absence as one state and carries no complete account of why an item is missing. The signed artifact adds survey-level information about the measurement process through the second channel introduced in Section 1.

The `surveyFit` object is designed to answer:

How much of the producer-reported applicable domain and architecture lens surface actually resolved in this scan?

The answer is carried in signed, readable form without entering the score.[2][5] Separating survey completeness from the numeric result is part of the current design.

5.1 `depthApplicable`

`depthApplicable` is the producer-reported denominator for survey completeness, not for any calculation inside the score. It counts the domain and architecture lenses that the producer says apply to the codebase.[2][5] Because applicability concerns the codebase rather than the catalogue alone, different repositories can legitimately have different denominators.

A lens judged inapplicable is excluded from the denominator. Applicability determines the survey surface before the separate question of whether measurement over that surface resolved.

For example, a codebase with no relevant accessibility surface should not have a hypothetical Accessibility lens counted as an unresolved applicable lens merely because Accessibility exists in the CAI lens catalog. Counting it would report a blind spot where there was nothing to see, and the completeness figure would fall for a reason unconnected to how thoroughly the survey was carried out.

Applicability is asserted by the producer from its model of the codebase and the analytical surfaces the codebase presents. CAI carries the value inside the producer's signed evidence, preserving both the claim and its provenance. The reference scorer does not validate whether excluded lenses were in fact inapplicable. Assurance about that judgement must therefore come from evidence outside the CAI fold.

This allocation of responsibility should remain explicit in report language. "Applicable depth: 8" means that the producer reported eight applicable lenses; it does not mean that the CAI scorer examined the codebase and independently arrived at eight. A recipient may accept, inspect or challenge that determination, but the number itself records provenance rather than independent agreement.

5.2 depthFired

`depthFired` counts the applicable domain and architecture lenses that produced a result.[2][5] It separates the surfaces the survey resolved from those it identified as applicable but did not resolve.

The derived difference:

```
depthApplicable - depthFired
```

is the producer-reported unresolved applicable surface at this level. It is a count of lenses: it identifies no individual lens, records no reason and makes no claim about what a resolved result would have shown.

The current evidence model exposes a derived `Abstained` property based on that difference, but `Abstained` is not a separately supplied field in the serialized artifact.[2] Two artifacts carrying the same two counts therefore carry the same information. An implementation treating `Abstained` as an independent producer value would assume a field the artifact does not contain.

If:

```
depthApplicable = 8
depthFired      = 5
```

then three applicable lenses did not resolve. The pair provides a quantitative but anonymous statement: it records neither which lenses remained unresolved nor why. The signed artifact can nevertheless expose a gap that the headline alone cannot show.

The anonymity of the count sets a hard reporting boundary. A report can state that three applicable lenses remained unresolved, but it cannot name them from `surveyFit` alone. Naming a particular unresolved lens requires another source that identifies it. The same applies to cause: the count does not distinguish unsupported analysis, an attempted analysis that produced no result or any other upstream explanation.

This count-level visibility is still an improvement over a headline without completeness context. It tells the recipient that the resolved evidence did not cover the whole surface the producer itself considered applicable. What it cannot do is turn that gap into a diagnosis, and the report should not use the presence of a count to imply that a reason was also recorded.

5.3 Nothing applicable and nothing resolved are different

The current implementation deliberately distinguishes:

```
depthApplicable = 0
depthFired      = 0
```

from:

```
depthApplicable = 6
depthFired      = 0
```

In the first case, the completeness fraction is undefined because its denominator is zero: no domain or architecture lens was reported as applicable. In the second, the completeness fraction is zero because six applicable lenses were identified and none resolved.[5] Rendering both cases merely as "zero resolved" would collapse the absence of an applicable surface into the failure to resolve an identified surface. `depthFired` therefore derives its meaning from `depthApplicable`, and the two counts must be displayed and interpreted together.

The distinction belongs specifically to the denominator. A zero numerator has no stable interpretation until the applicable count is known. Reporting only `depthFired = 0` would leave the reader unable to distinguish an empty applicable set from a wholly unresolved one. Any percentage or label derived from the counts must preserve the undefined case rather than coercing it into numerical zero.

5.4 Optional producer context

`surveyFit` may also include:

- a language applicability value;
- a producer-defined band;
- a producer explanation.[2][5]

These supplied values describe the producer's view of the survey. They are metadata rather than CAI scoring variables and have no defined effect on a lens result, band or headline.

A standards-facing report should identify that provenance, especially for the producer-defined band. A producer band and a CAI band are both labels attached to a result; rendering them without differentiation invites the unsupported inference that CAI derived both.

The producer explanation has the same provenance boundary. It may supply useful context for why the survey was limited or how the producer interpreted the counts, but it remains authored metadata rather than a CAI finding. Signing protects the text from undetected alteration after issue; it does not convert the explanation into a normative interpretation by the standard.

5.5 Signed, but not scored

The survey-completeness object is covered by the same signature as the scored evidence.[5] Completeness could have travelled outside that signature, but its inclusion makes subsequent alteration detectable.

Tests added with the feature establish two behaviours:

1. changing `surveyFit` changes the signed content and therefore is integrity-protected;
2. adding even the worst possible reported survey depth does not move the CAI headline or band.[5]

Together, these behaviours provide integrity without numeric influence. A recipient can verify that the completeness statement is the one the producer sent, while the scorer gives it no bearing on the accompanying result. Because the statement is part of the artifact rather than a companion document, it travels with the scored evidence even when a reporting surface chooses not to display it. A thin survey consequently lowers no lens, caps no band and changes no headline; narrow measurement is not recorded as worse code condition.

The combination also allows the two layers to evolve under different controls. A report can improve how it presents completeness without changing the scorer, while a proposal to make completeness score-bearing would require a change to the scoring method rather than a display adjustment. This separation helps reviewers identify whether a proposed change concerns communication, evidence semantics or the numeric standard itself.

6. Reporting survey completeness with the headline

Consider two reports produced under the same rubric, each carrying a signed evidence bundle. Both lead with the same number:

```
CAI 70
```

Report A also says:

```
depthApplicable = 8
depthFired      = 8
```

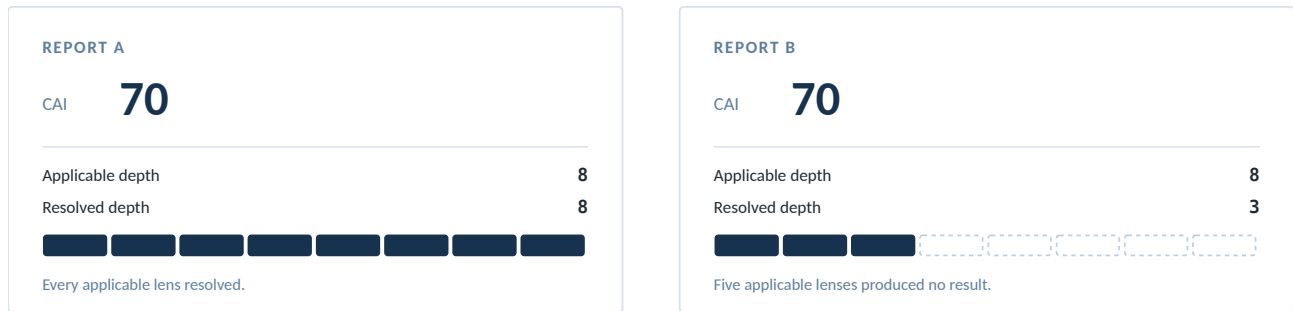
Report B says:

```
depthApplicable = 8
depthFired      = 3
```

Both headlines are valid outputs of the same fold, but they rest on surveys of different producer-reported breadth. Report A states that every applicable domain and architecture lens resolved. Report B's 70 was computed from the evidence that did resolve while five applicable lenses produced no result. A decision that treats the headlines as interchangeable assumes an evidential equivalence that neither report establishes.

The difference is not a correction to either score. Each headline is the result of applying the named rubric to its score-bearing evidence. What changes is the scope a recipient can reasonably attach to that result. Report A carries a producer claim of complete resolution over its applicable survey surface; Report B carries an explicit statement that most of its applicable lenses did not resolve.

Completeness should accompany rather than replace the headline.



Both headlines are valid outputs of the same fold under the same rubric. Quoted alone they are indistinguishable. Two lines make the difference visible, and the counts are producer-reported: they say how many, never which, and never why.

Figure 6. Equal headline, different survey breadth. Two valid results of the same fold under the same rubric, distinguished only by the producer-reported completeness counts beside them.

The score remains the output of the fold, while the two counts qualify the breadth of the producer-reported survey behind it. Presenting both facts allows a recipient to apply a decision rule without requiring CAI to merge them into a single scale.

6.1 Separating completeness from code condition

A lower score normally means that the scored code evidence was worse under the rubric. Reducing Report B's headline would instead express a conclusion about the codebase from a fact about the survey. The measurement process resolved less of the applicable analytical surface, but that fact says nothing about what a completed survey would have found. A recipient could not distinguish a thin-measurement penalty from a measured code finding if both lowered the same number.

The standard could deliberately define measurement incompleteness as part of codebase quality, but that choice would change the meaning of every headline and would require an explicit normative rule. The current design keeps code condition and survey completeness in separate fields so the artifact can distinguish a thin survey from poor measured code.

Such a rule would also need to define the rate and shape of the penalty. Treating every unresolved lens as equivalent would require evidence that the lenses carry comparable significance; weighting them differently would require an additional model. No such exchange rule exists in the current scorer. Lowering the headline without one would therefore introduce a discretionary policy into what is otherwise a deterministic fold.

The August 2026 `surveyFit` change explicitly states the opposite principle:

a thin survey is not a bad codebase.[5]

The scorer enforces this principle by ignoring `surveyFit`. Because the fold never reads the field, a reporting change cannot turn survey thinness into a numeric penalty without a corresponding change to the scorer.

6.2 Visibility of survey breadth

Identical headlines can otherwise be over-read as identical evidence claims. The score reports what the resolved, score-bearing evidence produced under the rubric and makes no claim about evidence that was absent. `surveyFit` reports the breadth of the producer-reported applicable domain and architecture survey surface, addressing a question outside the score's scope.

This division also clarifies what confidence a recipient may place in the arithmetic. A thin survey does not make the calculation over the available evidence less reproducible; the same evidence and rubric still produce the same result. It limits the breadth of the evidence to which that arithmetic was applied. Describing the issue as arithmetic uncertainty would therefore obscure the actual limitation.

Together, they allow a recipient to ask two separate questions:

What result did the evidence produce?

and:

How much applicable survey surface did the producer report as resolved?

Keeping the answers separate avoids inventing an exchange rate between unresolved lenses and points of code quality. No such relationship is established by the current model or evidence.

6.3 Reporting principle

Where `surveyFit` is available, report-facing CAI material should place it close enough to the headline to prevent a reasonable inference of complete survey depth from the headline alone. Completeness disclosed on a later page or behind a link cannot qualify a number that is read and repeated independently.

The display need not compete with the headline. A compact pair such as “8 applicable, 3 resolved” is sufficient to signal that a completeness statement exists and to prevent a recipient from assuming full depth. Detailed producer explanation can remain elsewhere, provided the count-level boundary remains visible where the score is first encountered.

The field should not be described as:

- code quality;
- a confidence interval;
- a probability that the CAI is correct;
- a scored CAI component;
- independently validated applicability.

It is producer-reported descriptive completeness information carried with the signed evidence.[2][5] The five excluded descriptions attribute to two counts something they cannot provide: a judgement about the code, a statistical guarantee, a role in scoring or independent validation. Within its actual scope, the field remains useful because it shows how much of the intended survey the producer says stands behind the headline, a question no other artifact field answers.

7. Artifact-level semantic boundaries

An account of CAI must use the states the artifact actually records. A richer measurement process may distinguish all of the following states for each dimension:

```
applicable
not applicable
attempted
not attempted
resolved
unresolved
unsupported
deliberately abstained
```

These distinctions could support a useful future design, and a producer may know which state applies while conducting the survey. The current CAI numeric evidence model does not record which state produced an omitted dimension. Using this vocabulary as though it described current CAI fields would misstate the standard and send implementers looking for data that is absent.

The restraint is about representation, not about whether the upstream distinctions are real. Measurement software may need to know that an analyser lacks language support, that a check was attempted but unresolved or that a criterion was excluded as inapplicable. Unless that state is carried into the CAI artifact under a defined schema, however, the CAI layer cannot expose it consistently or make conformance depend on it.

7.1 Dimension-level absence remains coarse

At dimension grain, several possible real-world causes collapse into one numeric state:

no scored item is present.

The artifact therefore supports neither a universal claim that every omitted dimension was explicitly adjudicated as inapplicable nor a universal claim that every omission was a producer abstention. Both readings attribute a decision or intent that the evidence did not record. The durable statement is narrower: the item is absent from the scored evidence, and its reason is either recorded elsewhere or unavailable.

Both unsupported readings can appear attractive for different reasons. Treating every omission as inapplicable makes the survey appear deliberately complete, while treating every omission as abstention makes the model appear more cautious than the fields demonstrate. Neither rhetorical advantage warrants expanding the semantics of the artifact. The paper uses the same neutral description for every dimension-level omission because that is the only description valid across all cases.

7.2 Survey-level applicability is richer, but scoped

`surveyFit` adds a higher-level distinction between producer-reported applicable and resolved domain and architecture lens depth.[2][5] It counts unresolved applicable surface without identifying individual lenses or explaining why they did not resolve. Its unit is the lens rather than the dimension, so it provides no reason taxonomy for a deterministic dimension missing from a category. The numeric fold and signed completeness object consequently operate at different grains: the former carries dimension absence as a single numeric state, while the latter carries two lens-level counts. Neither can answer the other's question.

A count at lens grain cannot be distributed back across the dimensions beneath those lenses. Three unresolved applicable lenses do not imply three missing dimensions, nor does a collection of missing dimensions reveal the `surveyFit` denominator. Any attempt to reconcile the two mechanically would need identifying relationships and reasons that the current count structure does not provide.

A third layer lies outside the single-survey artifact. Published corpus readings mark a figure whose population could not be fully observed as a `floor`, a lower bound carrying a count of unseen members. The corpus contract states that such a figure must not be summed with or ranked against a measurement.[12] This is machine-readable missingness for an aggregate over many surveys, not for one evidence bundle. It neither enters the fold nor supplies a reason for an absent dimension.

7.3 Producer terminology is not automatically CAI wire semantics

A producer may maintain richer internal states than the CAI artifact exposes. Earlier producer-side language-support work, for example, distinguished not-applicable analysis, abstention and analysis depth for its survey methodology.[6] These concepts are valid historical and methodological context, but a producer term becomes a CAI artifact state only when the CAI evidence model represents it. Standard-level terminology creates an implementation and interpretation commitment; terminology without a corresponding field implies semantics the artifact cannot deliver.

Historical producer terminology can still inform future design and explain where present concepts originated. Its use must remain attributed and scoped, however. Importing the vocabulary without the corresponding fields would make a producer's internal state model appear normative, while rejecting the vocabulary entirely would discard useful design history. The current wording preserves the context without claiming that CAI serializes those states.

The current founding paper therefore uses the more conservative CAI wording:

- omitted scored dimension;
- zero-confidence present dimension;
- partially covered dimension;
- null meta-dimension;
- absent lens;
- producer-reported applicable and resolved `surveyFit` depth.

Each term in this list names something carried by the artifact. Restricting the vocabulary in this way prevents a producer-specific internal state model from being silently promoted into CAI semantics.

7.4 A richer taxonomy would be a real standard change

If a future CAI version adds explicit machine-readable absence reasons, it will add semantic fields rather than merely improve documentation. The rest of the system will have to agree on the meaning and use of those fields.

It could affect:

- reporting semantics;
- conformance expectations;
- comparison rules;
- challenge paths;
- future eligibility or scoring rules;
- historical interpretation.

Such a change requires deliberate design and versioning because a new reason field would immediately be treated as authoritative in reporting, conformance, comparison, challenge and historical interpretation. The account in Sections 2 to 6 is complete with respect to the states the current artifact records. Additional provenance would be a change to the standard rather than an editorial elaboration of the present one.

Versioning would also protect historical artifacts. A recipient must be able to tell whether the absence of a reason means that the producer supplied none under a schema that supported reasons or that the artifact predates the field entirely. Without that distinction, a new taxonomy could retroactively change the apparent meaning of old omissions. The schema, conformance rules and report rendering would therefore have to move together.

8. Four counterarguments

8.1 “Just score missing evidence as zero”

Scoring missing evidence as zero would make every rubric dimension carry a value, every category use the same nominal set and every report contain the same number of inputs. It would also give zero two meanings: the lowest result on the measurement scale and a marker that no measurement occurred.

The fixed-set property has an obvious appeal. Two category results would always share a denominator, and omission could no longer raise an average by removing a weak contributor. The price is semantic rather than arithmetic: the number zero would stop identifying a measured outcome and would instead conflate measurement with the absence of measurement.

A measured zero says:

the producer measured this criterion and supplied the lowest score.

A missing item says:

there is no score-bearing item here.

Converting every absence into zero would create failure evidence where no measured failure was supplied. A recipient could not distinguish failed criteria from unexamined ones. The rule would also penalise irrelevant surfaces unless another mechanism had already resolved applicability accurately enough to remove every irrelevant criterion. CAI keeps absence and measured zero separate, accepting the set effects described in Section 3 so that every numeric contribution traces to supplied measurement evidence.

Perfect applicability resolution cannot simply be assumed as a prerequisite because the current dimension-level artifact has no complete reason taxonomy for absence. A zero-fill policy would therefore rely on a distinction that the fold cannot verify while erasing the visible difference between a supplied zero and a generated one. The resulting arithmetic would be simpler but the evidence claim would be less precise.

8.2 “If it is absent, it has no effect”

An absent item contributes no direct numeric value. The surviving fold can nevertheless change because the result is a property of the aggregated set rather than of the missing item alone.

The apparent contradiction disappears once the level of the claim is specified. At item level, the missing dimension contributes nothing. At aggregate level, the collection of contributors has changed, so the calculation answers a question about a different set. Saying that the item has no value is correct; extending that statement to say the result cannot change is not.

The category example in Section 3 moves from 50 to 80 when the weak contributor disappears and from 50 to 20 when the strong contributor disappears. Neither movement came from a value the missing dimension contributed, since it contributed none. Both came from the mean being taken over a different set.

The synthetic lens example moves from 57.07 to 77.10 when the weakest lens disappears. At the headline layer, both the set and the ordered weights are recomputed.

The correct statement is:

Absence contributes no value, but changing the set being aggregated can change the result.

8.3 “Coverage already tells us completeness”

Coverage and `surveyFit` both concern measurement reach but operate at different grains and answer different questions.

Coverage belongs to an individual measured deterministic dimension and changes its effective score:

$$\text{effective} = \text{score} \times \text{coverage}$$

`surveyFit` is producer-reported descriptive information about applicable versus resolved domain and architecture lens depth. Its unit is the lens and its scope is the survey rather than an individual measured value.

Coverage is score-bearing: it is applied before the category mean and can take a contributor below the critical gate. `surveyFit` is non-scored, so omitting it from a report does not alter the meaning of the calculated number but suppresses a separate fact about survey breadth. Full coverage on every present dimension implies nothing about how many lenses resolved, while complete survey depth implies nothing about how thoroughly any individual criterion was measured.

The fields can therefore move independently. A scan may report full coverage for every dimension it contains while resolving only a small share of the applicable lens surface. Another may resolve every applicable lens while individual dimensions inside those lenses have partial coverage. Neither pattern is contradictory, because the fields describe different units and different parts of the measurement process.

8.4 “A thin survey should score lower”

The objection identifies a genuine decision problem: a recipient with less evidence is in a weaker position than one with more. Lowering the CAI score would convert that limitation in the recipient's knowledge into evidence of poor software, however, while the current design keeps the two claims separate.

The CAI artifact does not establish why a survey was thin. That cause belongs upstream in the producer's measurement evidence. Reduced survey depth alone therefore supplies no code-condition finding, and a numeric penalty would treat every gap alike without evidence that the treatment was warranted.

A recipient may reasonably demand broader evidence before accepting the result for a particular use. Until the standard deliberately incorporates such a requirement into the scoring definition, the resulting decision rule belongs to the use of the score. Keeping that rule explicit distinguishes a codebase judgement from an evidence-sufficiency judgement.

The published corpus reading is consistent with keeping these claims separate. Across the seventeen language fields reported on 14 September 2026, median score does not move monotonically with typical resolved depth: Swift has a median of 52.8 at depth 9, Erlang 53.2 at depth 10, Kotlin 53.2 at depth 13, Python 53.0 at depth 15 and C# 49.7 at depth 26.[11] Depth is available for 2,585 of the 3,739 projects in those language fields. Because these are field-level observations from one day's reading rather than a controlled project-level study, they cannot establish the relationship between score and depth for individual projects. They do show that the published field data provide no basis for mechanically treating a thinner survey as a poorer codebase.

The observation must remain within its stated grain. The language-field medians combine many projects, and the typical depth values are not paired project-by-project with those medians in the comparison above. Confounding differences among languages, repositories and measurement support remain uncontrolled. The figures support only the limited conclusion used here: greater typical resolved depth is not accompanied by a simple monotonic increase in the published field medians.

The current CAI approach exposes survey thinness separately through the two counts described in Section 6. A decision-use framework may still treat a thinly surveyed result as insufficient for procurement, review or an internal threshold. Such a rule would make the recipient's evidential requirement explicit without changing the CAI score into a claim it was not designed to make.

9. Empirical robustness under missingness

The scorer establishes how the current method handles missingness, and the claims in Sections 2 to 8 can be checked against the published fold. Measuring how much those rules affect real results requires evidence from repositories rather than further inspection of the implementation.

A stronger future claim might be:

CAI scores are robust to realistic patterns of omitted and partially covered evidence.

This claim concerns a distribution of real repositories and therefore requires empirical support beyond a scorer, rubric or specification. The published corpus readings record 3,957 measured codebases as of 14 September 2026, with per-project resolved depth available for a growing subset and counts published behind each share.[11][12] The corpus provides a population for a study, but no perturbation analysis over that population has yet been published.

The distinction between an available corpus and a completed study matters. Corpus size alone says nothing about how scores respond when evidence is deliberately removed or coverage is varied. The study must define which perturbations represent plausible measurement conditions, run them against frozen inputs and report the distribution of movement rather than selected examples. Until then, the corpus demonstrates research feasibility, not robustness.

9.1 What a corpus study should test

A missingness study should freeze the repository corpus, rubric and scorer before perturbing evidence in controlled ways. Holding all three constant is necessary to attribute score movement to the perturbation rather than to a change in scoring rules or implementation.

The protocol should also retain the unperturbed evidence for each repository as its baseline. Each modified bundle can then be scored against the same rubric and scorer, allowing headline movement, band movement and loss of lens participation to be attributed to a specified intervention. Results should be reported across repository classes and baseline evidence profiles so that an average effect does not hide concentrated instability in a smaller group.

At minimum, the study should measure effects from the dimension layer through the headline, then test whether those effects are visible to a report reader:

1. headline movement when individual scored dimensions are omitted;
2. headline movement when categories become unscored;
3. headline movement when measured lenses disappear;
4. band changes caused by omission rather than repository change;
5. changes when coverage is progressively reduced;
6. cases where reduced coverage triggers or removes a critical gate;
7. differences between omission and partial coverage for the same criterion;
8. interactions between model-aware lens applicability and headline composition;
9. whether `surveyFit` makes materially thin evidence obvious to report readers;
10. rank-order movement if CAI is ever used for comparison across multiple repositories.

9.2 Sensitivity analysis method

The OECD/JRC handbook treats indicator inclusion and exclusion, missing-value treatment, weighting and aggregation as model assumptions suitable for uncertainty and sensitivity analysis.[8] CAI makes a defined choice in each category. Current JRC guidance likewise identifies missing data, weights and aggregation systems as assumptions that can influence the message carried by a composite indicator.[10] Sensitivity analysis is therefore an appropriate method for estimating how much CAI output depends on changes to the evidence set and related modelling assumptions.

The proposed analysis should vary one assumption or evidence condition at a time where possible, then examine interactions explicitly. Removing a dimension, removing all contributors to a category and removing a lens are related interventions, but they act at different layers of the fold. Progressive coverage reductions add a different path because the item remains present and may cross the critical threshold. Reporting them separately preserves the mechanism behind each observed movement.

This methodological support does not imply that CAI should adopt an OECD or JRC imputation technique. Imputation fills a gap with an estimate, while the current CAI rule leaves an absent item without an estimated value. The rule is already defined; the proposed study would measure its consequences at scale rather than replace it.

The external methodology supports the research question:

How much does the output depend on plausible variations in the evidence set and missingness assumptions?

9.3 Established behaviour and open empirical claims

The current scorer, evidence model and tests establish that:

- omitted items do not receive a synthetic zero;
- confidence zero contributes no numeric category weight;
- partial coverage reduces effective score;
- partial coverage can interact with the critical gate;
- absent primary-path lenses leave the headline fold;
- `surveyFit` is descriptive and non-scored;
- changing the measured set can move a composite result.

These are implementation and mathematical facts.[1][2][5] They can be checked in the scorer or reproduced arithmetically. A corpus study is required for stronger statements about:

- typical effect size;
- frequency of material movement;
- robustness across repository classes;
- applicability accuracy;
- probability of band changes;
- whether the current rules are empirically optimal.

The two lists separate the defined rule from its empirical consequences. Repository evidence is unnecessary to establish the first list and indispensable for the second. The absence of a corpus study should therefore constrain claims about effect size, frequency, robustness and optimality without weakening verified statements about current implementation behaviour.

This division also sets the publication boundary for later work. A verified scorer change belongs in the account of current behaviour once the reference state moves. A finding about typical movement belongs in the empirical record only after the corpus, perturbation protocol and analysis are available for review. Keeping those evidence types separate prevents implementation documentation from being mistaken for validation research.

10. A reporting model for missing evidence

The practical reporting requirement is to preserve distinctions already present in the artifact rather than calculate another summary number. The interpretive problems described in Sections 2 to 9 arise when a reporting surface fails to carry those distinctions through to the recipient.

A useful report sits between a bare headline and the complete evidence bundle. The headline alone hides the measured set and survey breadth; the full artifact may be too detailed for the initial decision surface. The reporting elements below select the fields needed to understand what participated in the fold, what limited the reported bands and how broad the producer says the applicable survey was.

A report can expose:

Reporting element	What it tells the recipient
Headline CAI	Result of the current scored evidence under the named rubric
Lens results	Which lenses participated and how they scored
Category breakdown	Which category results contributed inside each lens
Critical contributors	Which measured low-effective contributors triggered a lens band cap
Rubric version	Which scoring semantics governed the result
<code>surveyFit.depthApplicable</code>	Producer-reported applicable domain/architecture lens count
<code>surveyFit.depthFired</code>	Producer-reported resolved count within that applicable surface
Producer explanation	Human-readable context for the completeness result, when supplied
Limitations	What the artifact does not establish

These elements form a selective report rather than a rendering of the entire artifact. Together they allow the report to distinguish:

This dimension failed.
This dimension was only partially covered.
This dimension is absent from the score-bearing evidence.

At the `surveyFit` level, the guaranteed statements are count-level:

The producer reported eight applicable domain/architecture lenses, of which five resolved.
Three applicable lenses remained unresolved.

Specific claims about which individual lens applied, failed to resolve, or sat outside the applicable denominator require additional identifying evidence beyond the `surveyFit` count structure itself. That is the boundary of what this section proposes: a report can say how many, and a report that says which one has taken that from somewhere else.

The additional evidence may legitimately exist in a producer report or measurement trace. If a reporting system uses it, the source should remain identifiable so that the count-level CAI statement is not confused with a richer producer statement. This preserves the ability to challenge a named-lens claim without disputing what the signed `surveyFit` counts actually contain.

The three dimension-level statements identify a criterion, whereas the survey-level statements provide counts. Mixing the two registers would imply that `surveyFit` identifies lenses when it only counts them. Collapsing these states into one blank cell, N/A marker or generic zero discards the distinction among a measured zero, an omitted dimension, a partially covered dimension and an unresolved lens.

Those display shortcuts are especially hazardous when reports are exported or quoted outside their original interface. Colour, tooltip text or page-level explanation may disappear while the cell value remains. Explicit labels for measured zero, partial coverage, absence and unresolved survey depth preserve their meaning in a way that a generic placeholder cannot.

10.1 Equal score, different evidential breadth

A useful report example is the one Section 6 argued for, rendered as a recipient would meet it:

Report A	
CAI	70
Applicable depth	8
Resolved depth	8

Report B	
CAI	70
Applicable depth	8
Resolved depth	3

Quoted alone, the two headlines are indistinguishable. The arithmetic has the same status in both cases, but the producer-reported survey breadth differs. Two additional lines are sufficient to expose that difference. Keeping them adjacent to the headline preserves both facts at the point where a reader interprets the result.

The example does not prescribe a universal visual design. It establishes the minimum semantic content: the applicable and resolved counts must remain associated with the headline and retain their producer-reported status. A graphical indicator, fraction or short sentence may carry the same information if it also preserves the undefined-denominator case described in Section 5.3.

10.2 Completeness is not a second CAI

`surveyFit` is a descriptive completeness surface rather than a shadow score. Its two counts, their difference and optional producer context answer one question about applicable and resolved survey depth. Using the field to address broader uncertainty or evidence-strength questions would assign semantics unsupported by its data.

In particular, no threshold in the current standard turns a given resolved fraction into a second pass/fail result. A consumer may define such a threshold for its own use, but it should be labelled as a decision rule rather than as another CAI score or band. Otherwise the report would make a consumer policy look like a property of the standard.

A future standard may define richer coverage, uncertainty or evidence-strength constructs. As Section 7.4 explains, these would be versioned additions rather than reinterpretations of `surveyFit`. Any such construct should answer a defined decision problem and state its relationship to the existing headline and completeness fields.

That discipline protects both fields from semantic drift. The headline remains a result of score-bearing evidence under a named rubric, and `surveyFit` remains a producer-reported account of applicable versus resolved survey depth. New constructs can then be evaluated on their own definitions instead of being attached informally to one of the existing numbers.

11. Limitations and maturity boundary

This paper describes current CAI behaviour for missing and partial evidence. It does not establish that the policy is the only reasonable design. The present technical sources establish the following implementation claims:

- a present score-bearing zero is different from an omitted scored item;
- omitted deterministic dimensions do not receive synthetic zero values;
- zero-confidence dimensions have no numeric category weight and do not trigger the critical gate;
- partial coverage reduces the dimension's effective score;
- low effective scores can trigger the rubric's critical gate;
- null meta-dimensions do not participate;
- absent lenses in the primary evidence path do not participate in the headline;
- the surviving primary-path lens fold is recalculated over the measured set;
- `surveyFit` carries producer-reported applicable and resolved domain/architecture depth;
- `surveyFit` is signed, descriptive and non-scored;
- the reference scorer does not use `surveyFit` to change the headline or band.[1][2][5]

Several broader questions remain open.

Dimension-level absence reasons

The current numeric artifact preserves no complete reason taxonomy for omission. Adding one would require deliberate schema and semantic design, together with versioned rules for how implementers and recipients use the new states.

Applicability accuracy

The producer reports applicability at the `surveyFit` level, but no measured precision or recall rate has been established for those decisions. A validation study would require a labelled corpus and a defined adjudication protocol.

Robustness to missingness

The scorer defines how omission behaves but provides no distribution of omission effects across a repository population. The corpus readings now publish a population over which a perturbation study could run and record resolved depth for individual projects, but no such perturbation result has been published.[11][12]

Two published depth figures

The corpus reading's depth figure counts the index dimensions that resolved on a project. `surveyFit.depthApplicable` and `depthFired` count domain and architecture lenses. Public CAI surfaces call both figures depth, but they operate at different grains and cannot be compared as though they measured the same population.[5][11]

Partial coverage policy

The current implementation establishes the `score × coverage` rule and its interaction with the critical gate. No empirical evidence presented here establishes that policy as the optimal treatment of partial evidence.

Comparability

A CAI result should be interpreted with its rubric, subject, measured lens set and available completeness context. An identical headline does not establish direct comparability between repositories measured over different surfaces.

Producer neutrality

`surveyFit` is producer-reported. CAI carries and integrity-protects the values in signed evidence but does not independently validate the producer's applicability and resolution claims.

These limitations mark the boundary between the behaviour established by the current architecture and the claims that require later research or standard evolution.

12. Conclusion

The current CAI implementation distinguishes measured values from missing evidence at each layer of the fold. A measured zero remains score-bearing; partial coverage reduces a present dimension's effective value; zero-confidence dimensions, null meta-dimensions and absent primary-path lenses contribute no numeric value at their respective layers. Separately, signed `surveyFit` counts describe producer-reported applicable and resolved domain and architecture survey depth without lowering the codebase score.

These distinctions preserve the difference between a measured failure and an unavailable measurement, while giving recipients a way to see that two identical headlines may rest on surveys of different breadth. They do not make omission arithmetically neutral:

Omission is not a numeric penalty, but omission can still move the result because it changes what is being aggregated.

A report should therefore carry the measured lens structure and available completeness context beside the headline. The current model can support that reporting without inventing failure values for gaps or merging survey reach into code condition.

The remaining limits are explicit. The artifact has no complete dimension-level reason taxonomy, the scorer does not validate producer-reported applicability, and the distribution of omission effects across repositories has not been established. A corpus-level perturbation study can test how robust results are under realistic missingness patterns. That open empirical question limits claims about robustness and optimality; it does not prevent an accurate account of the current implementation.

References

CAI technical references are pinned to the implementation state reviewed for this draft, through commit `1d557c36faa6f7650cfe6746db01665ad2ac9bfc`, unless a more specific commit is named. External references support general composite-indicator methodology only. They do not define CAI semantics or validate the CAI method by analogy.

Corpus references 11 and 12 cite dated readings. The published corpus grows between readings and the cited addresses serve the current reading, so the figures drawn from them are reported as of their stated reading date rather than as values a reader will find at those addresses today.

1. **CanineCC/CAI**, *CaiScorer.cs*. Current category, lens and headline fold, confidence weighting, coverage interaction, critical gate, OWA treatment and thin-lens fallback.
<https://github.com/CanineCC/CAI/blob/1d557c36faa6f7650cfe6746db01665ad2ac9bfc/src/Cai.Scoring/CaiScorer.cs>
2. **CanineCC/CAI**, *EvidenceBundle.cs*. Current evidence fields, coverage, confidence, advisory status, omitted-field semantics and `SurveyFit` model.
<https://github.com/CanineCC/CAI/blob/1d557c36faa6f7650cfe6746db01665ad2ac9bfc/src/Cai.Scoring/EvidenceBundle.cs>
3. **CanineCC/CAI**, *Lenses.cs*. Current ten-lens catalog and core/model-aware classification.
<https://github.com/CanineCC/CAI/blob/1d557c36faa6f7650cfe6746db01665ad2ac9bfc/src/Cai.Scoring/Lenses.cs>
4. **CanineCC/CAI**, *ScoringParameters.cs*. Current rubric-carried critical gate and other score-moving aggregation parameters.
<https://github.com/CanineCC/CAI/blob/1d557c36faa6f7650cfe6746db01665ad2ac9bfc/src/Cai.Scoring/ScoringParameters.cs>
5. **CanineCC/CAI**, commit `fc616ce3bf981194e0e351547a234aa6b22b3645`, 22 August 2026, *a signed CAI now says how complete the survey behind it was*. Primary source for signed `surveyFit`, its fields, its producer ownership, its non-scored status and the test that it cannot move the headline. <https://github.com/CanineCC/CAI/commit/fc616ce3bf981194e0e351547a234aa6b22b3645>
6. **Canine Development**, *Beyond the Language Logo*, W2 v0.11. Historical producer-methodology source for distinctions among language applicability, not-applicable analysis, abstention and survey-specific depth. Used as predecessor context only; current CAI wire semantics are taken from the CAI repository rather than imported from the producer's own terminology.
7. **Jimmy Borch / Canine Development**, *How the Codebase Assurance Index is constructed*. Founding construction paper, especially the current scoring path, coverage rule, omission semantics and `surveyFit` reporting boundary.
8. **OECD / European Union / European Commission Joint Research Centre**, *Handbook on Constructing Composite Indicators: Methodology and User Guide*, OECD Publishing, 2008, DOI `10.1787/9789264043466-en`. Used for the general proposition that missing-data treatment, weighting, aggregation and indicator inclusion/exclusion are consequential modelling choices in a composite indicator.
9. **European Commission Joint Research Centre**, *Step 3: Imputation of missing data*, Competence Centre on Composite Indicators and Scoreboards. Used only for the general proposition that missing values can affect the reliability and relative standing of composite indicators.
https://knowledge4policy.ec.europa.eu/composite-indicators/toolkit_en/navigation-page/10-step-guide_en/step-3-imputation-missing-data_en
10. **European Commission Joint Research Centre**, *Step 8: Sensitivity analysis*, Competence Centre on Composite Indicators and Scoreboards. Used for the general methodological role of sensitivity analysis in testing the influence of missing data, weights and aggregation assumptions.
https://knowledge4policy.ec.europa.eu/composite-indicators/toolkit_en/navigation-page/10-step-guide_en/step-8-sensitivity-analysis_en
11. **Codebase Assurance Index**, *Measured projects*, corpus reading of 14 September 2026, retrieved 14 September 2026. Per-language score spread and resolved survey depth over 3,907 measured projects in 17 language fields, depth defined as the number of the index's dimensions that resolved on a project and recorded for 2,585 of 3,739 projects in those fields. Readings are append-only and each is an observation of a single reading. The address serves the current reading, so the figures cited here are those recorded on the stated date and are not reproduced by the live page.
<https://codeassuranceindex.info/surveys/>
12. **Codebase Assurance Index**, *The state of the corpus*, corpus reading of 14 September 2026 taken 00:15 UTC, retrieved 14 September 2026. Aggregate reading over 3,957 measured codebases, published with the counts each share was derived from. Primary source for the stated contract that a share travels with its denominator and that a `floor` figure is a lower bound carrying an `unseen` count. The address serves the current reading, so the figures cited here are those recorded on the stated date and are not reproduced by the live page.
<https://codeassuranceindex.info/state-of-the-corpus/>

Verification record

Verified: 17 September 2026 against the CAI reference implementation at the commit named in the technical reference state above.

Every statement of scorer behaviour in this paper was rechecked at that date against the repository's current state. The fold, its constants and the `surveyFit` evidence model were unchanged between the pinned commit and the state on the verification date: no file under `src/Cai.Scoring/` was modified in any commit between them. The within-lens and across-lens OWA decays, the shipped-weight tolerance and its condition, the effective-value rule, the confidence-weighted category formula, the critical-gate threshold and its exclusion of zero-confidence and advisory contributors, the dimension count reported by a category result, and the absence of any reference to `surveyFit` in the scorer all hold as described.

The rubric line moves independently, and considerably faster. Fifteen rubric versions were published between the pinned commit and the verification date, the most recent being `rubric-2026.09.12`. Catalogue size moves with them: one release is recorded as changing from 154 entries to 152, and the catalogue current at the verification date carries 159 scored dimensions across ten lenses. No claim in this paper depends on a catalogue size.

Scoring semantics travel with the rubric version a result was produced under. Every statement of current behaviour here should therefore be read as describing the implementation on the verification date, and any particular result should be read against the rubric version that result names rather than against the values quoted in this paper. A reader returning to this paper after the reference state has moved should treat it as a dated account rather than as a description of whatever the implementation does at the time of reading.