

ASSURANCE

The Right Deliverable, Not Merely a Deliverable

Target, realization, divergence, and acceptance in Sawbill

How Sawbill establishes that a result matches the intent, target, policies, and expected evidence.

READERS

Engineering executives and leaders / Architects / Quality and security leaders / Risk and compliance leaders

Contents

The issue in one minute	3
What "conformance" means here	3
Three artifacts that must not be confused	3
The target graph	4
The actual graph	4
The divergence report	4
Alignment cannot be inferred from a name	4
Observation is a governed activity	5
Assurance spans the existing delivery chain	6
UNKNOWN is information, not a cosmetic failure	6
Tests are bounded evidence	7
Mechanical and semantic: two forms of evaluation	8
From divergence to the next work	8
Acceptance binds exact versions	8
Normative conformance is a profile, not the entire engine	9
Scenario: does the data export actually conform to the target?	9
Enterprise value	9
Fewer false positives in delivery	9
Explainable quality	9
Integrable regulatory compliance	10
Continuous drift management	10
Official references	10

The Right Deliverable, Not Merely a Deliverable

The issue in one minute

An agent can complete its task, produce valid code, and pass every available test without delivering what was actually requested. A requirement may be missing. The target architecture may have been bypassed. A configuration may not have been observed. A test result may belong to an older version.

Sawbill calls **deliverable assurance** the ability to connect:

an authorized intent, a versioned target, an observed realization, applicable policies, and the evidence required for acceptance.

Regulatory compliance may contribute to this assurance, but it is only one possible dimension.

What "conformance" means here

In everyday language, conformance first evokes laws, regulations, and standards. For an engineering product, the initial question is broader: does the deliverable match the contract that the organization chose to apply to it?

That contract may include:

- functional behavior;
- architecture, interfaces, and dependency constraints;
- security invariants;
- quality or performance objectives;
- a verification process and separation of duties;
- contractual obligations;
- controls derived from applicable standards or regulations;
- evidence and acceptance conditions.

Sawbill does not measure a universal quality. It evaluates correspondence relative to an identified target and assurance profile.

Three artifacts that must not be confused

The core of the approach relies on three distinct, versioned objects.

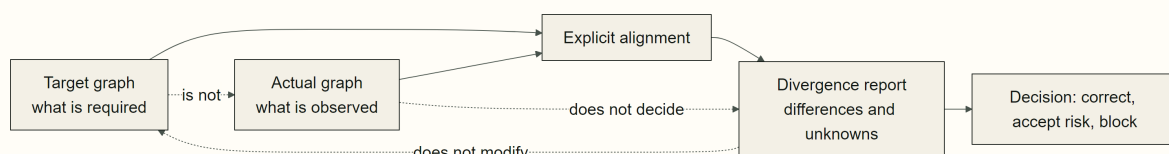


Figure 1 - Sawbill

The target graph

The target graph represents what is required or intended: capabilities, components, interfaces, rules, requirements, and relationships. It is bound to the versions of the sources, policies, packs, and profiles used to interpret it.

An intent or source document is not automatically an admitted target. Sawbill preserves the difference between a proposed target and the version that actually serves as the evaluation contract.

The actual graph

The actual graph describes what was observed at a given cutoff. It can include:

- code structures and dependencies;
- interfaces and configurations;
- deployment topology and artifacts;
- behavior observed through tests or traces;
- attributed functional claims;
- elements that were inaccessible, unattempted, or unknown.

The actual is not inferred from the target. An expected but unobserved component does not silently disappear; it becomes an established absence or an unknown, depending on available coverage.

The divergence report

The report compares an exact target with an exact realization under a given policy. It retains differences, ambiguous mappings, and unknowns.

Accepting a risk does not turn a divergence into conformance. Sawbill can retain both the technical fact - a difference exists - and its governance disposition - it has been temporarily accepted by a particular authority.

Alignment cannot be inferred from a name

Two objects with the same name, path, or partial digest do not necessarily play the same role. Sawbill makes alignment explicit between target and actual nodes.

An alignment relationship may be:

- established;
- a candidate;
- rejected;
- ambiguous;
- unknown;
- obsolete.

Only a relationship established under the applicable profile can feed subsequent deterministic rules. N:M relationships remain possible: one requirement may be realized by several components, and one component may contribute to several requirements.

Observation is a governed activity

An analysis is reliable only if its scope is defined before the result is seen. Otherwise, a tool can simply shrink the scope when part of the system is hard to inspect.

Sawbill uses an observation mandate that specifies:

- the expected population;
- sources and methods;
- dimensions to examine;
- required access rights;
- cutoff and data constraints;
- expected results and coverage limits.

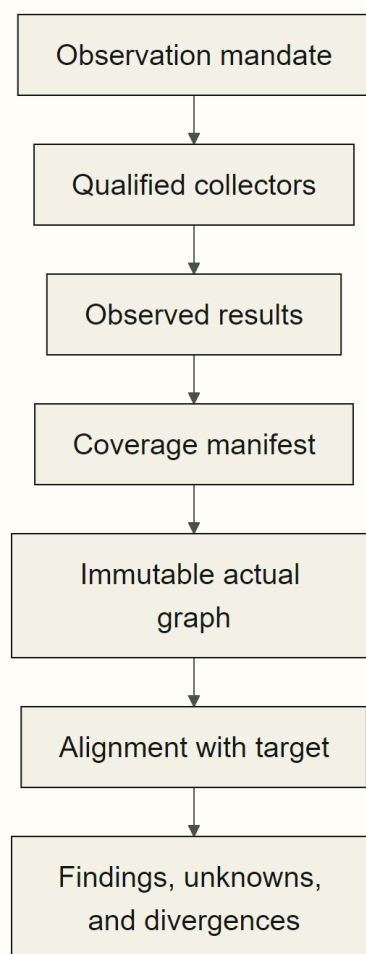


Figure 2 - Sawbill

A timeout, inaccessible source, partial partition, or method unable to detect a class of defect remains visible in the manifest.

Assurance spans the existing delivery chain

Sawbill does not replace tests, scanners, CI, artifact repositories, or observability platforms. It gives each an explicit place in an assurance case: every tool observes a defined subject; Sawbill links that observation to the target, coverage, and the decision that can actually use it.

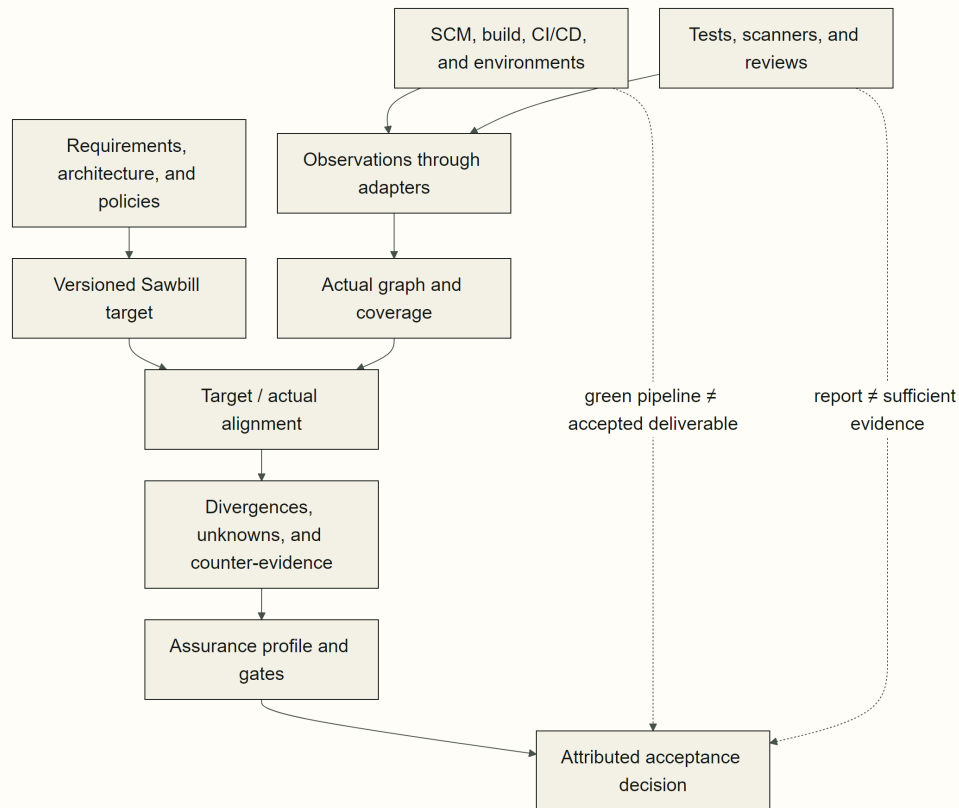


Figure 3 - Sawbill

The boundary moves from tool status to the demonstrated relationship between a target and an observed state. Tools retain their specialization; none can decide alone that the complete deliverable is acceptable.

UNKNOWN is information, not a cosmetic failure

Much software risk comes from positive conclusions drawn from an incomplete population. A scanner may cover 100 percent of the files it received while ignoring a runtime, configuration, or adjacent repository.

Sawbill therefore distinguishes process coverage from adequacy of the universe:

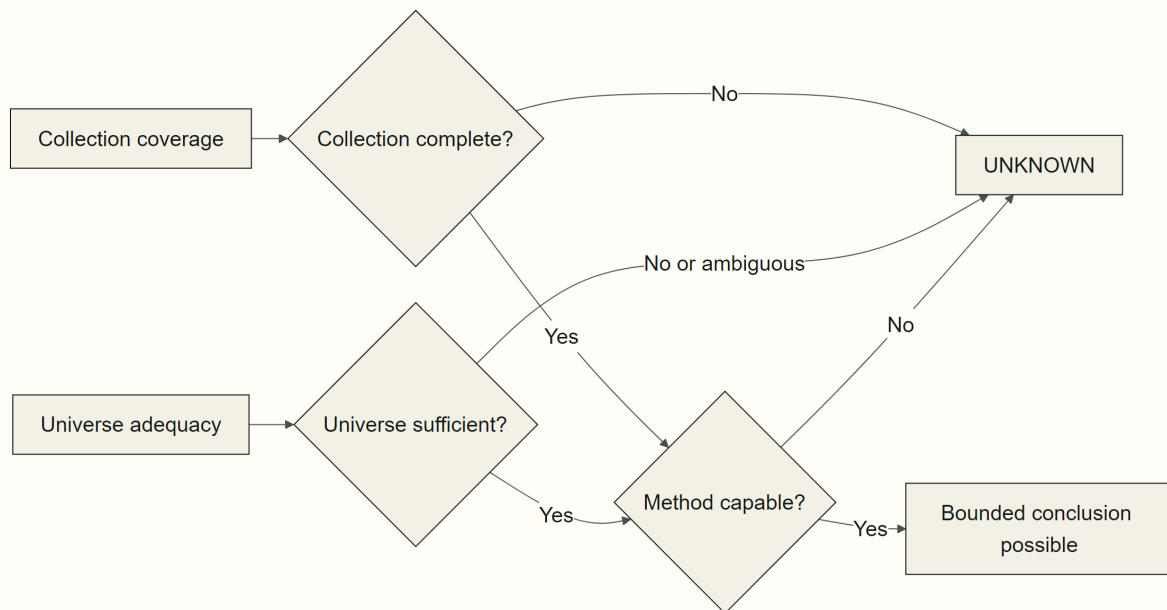


Figure 4 - Sawbill

A conclusion of absence requires a closed population, a capable method, a complete traversal, and an explicit negative result. Without them, Sawbill keeps UNKNOWN instead of producing a reassuring score.

Tests are bounded evidence

A unit test, integration test, and regression test do not answer the same question. Sawbill binds each test to a profile declaring its environment, data, subject, expected population, and oracle.

For a logical block, a verification set groups the exact packages involved:

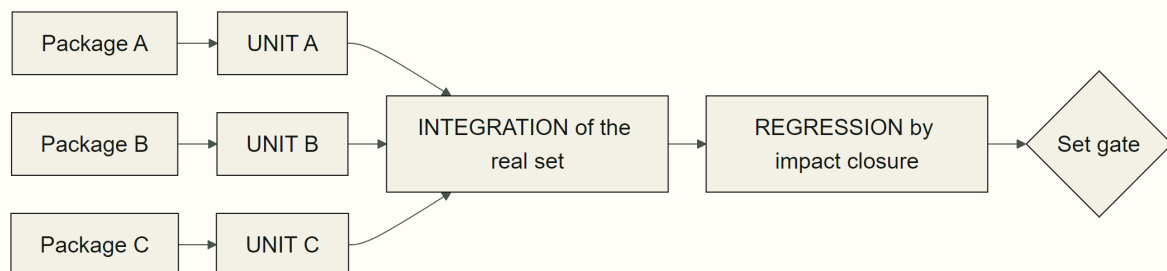


Figure 5 - Sawbill

A terminal step does not compensate for a missing step. A green unit test does not prove integration. An integration test does not automatically cover affected consumers. An N/A exemption must itself be justified, bounded, and authorized.

Mocks and simulators remain useful, but they prove only behavior under substitution. They do not demonstrate the behavior of the real provider or an external effect.

Mechanical and semantic: two forms of evaluation

Some questions have a deterministic oracle:

- do these bytes have this digest?
- is the schema valid?
- is this prohibited dependency present?
- does the signature match the expected bytes?

Others require semantic judgment:

- does the implementation actually realize the business intent?
- is the observation universe sufficiently complete?
- does the architecture address the risk in context?
- does a regulatory requirement apply to this organization?

Sawbill automates mechanical predicates and attributes semantic judgments. An agent can read, compare, and search for counterexamples, but its confidence score does not become mathematical proof. Material decisions may require independent or human review.

From divergence to the next work

The divergence report does not directly modify the system. It feeds an impact analysis, followed by a planning basis and a distinct execution DAG.

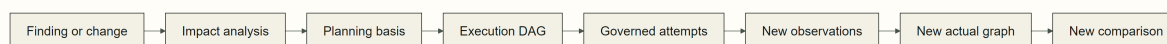


Figure 6 - Sawbill

This cycle avoids two shortcuts: automatically turning every divergence into a task, or treating a completed task as proof of realization.

Acceptance binds exact versions

An acceptance decision applies to a precise subject and a precise set of inputs: target, realization, policies, gates, and evidence. It is not a durable Boolean attached to a project name.

If code, policy, evidence profile, or coverage changes, the system does not silently reuse the old acceptance. It reevaluates usability and opens a successor when necessary.

Outcomes remain distinct:

Outcome	Meaning
ACCEPTED	The competent authority accepts this exact subject with these exact inputs
REJECTED	The decision is explicitly negative
BLOCKED	A mandatory condition prevents progress
UNKNOWN	The case is valid but insufficient to conclude

UNKNOWN does not mean low risk. It means the system refuses to manufacture certainty.

Normative conformance is a profile, not the entire engine

External obligations enter Sawbill as identified and versioned sources. A mapping links an external requirement to Sawbill controls, subjects, and evidence without copying or replacing the normative text.

The standards-first path is:



Figure 7 - Sawbill

Applicability, tailoring, and legal judgments remain attributed. A successful pack may contribute to a compliance case; on its own, it is not a certification.

Scenario: does the data export actually conform to the target?

The target requires:

- an export accessible only to the data owner;
- logging without sensitive content;
- processing in the authorized environment;
- expiry of the download link;
- unit tests, integration with real storage, and regression over existing access paths.

Observation finds the endpoint, access controls, and unit tests. It cannot, however, access the staging environment configuration. Real storage has not been exercised.

Sawbill concludes neither "conformant" because CI is green nor "non-conformant" by intuition. It produces:

- established alignments for the endpoint and access control;
- a divergence if logging exposes data;
- UNKNOWN for the inaccessible environment;
- an unsatisfied integration obligation for real storage;
- an impact analysis over existing access paths.

The decision maker knows exactly what is demonstrated, contradicted, or still unknown.

Enterprise value

Fewer false positives in delivery

Task completion, a commit, or a green pipeline does not replace comparison between intent and realization.

Explainable quality

Each conclusion retains its target, scope, method, coverage, and limits. Teams can debate the right profile rather than an opaque score.

Integrable regulatory compliance

Normative obligations become versioned sources and mappings within the same assurance path, without contaminating the engine with implicit legal interpretation.

Continuous drift management

Because target, actual, and divergences are distinct, Sawbill can detect change and reevaluate acceptance without rewriting history.

Official references

Source	External status	Contribution used	Limit retained by Sawbill
ISO/IEC/IEEE 29148:2018 - Requirements engineering	Published standard, confirmed in 2024 and under revision	Reference for requirements engineering and its information items	Licensed text is not reproduced; applicability remains explicit
ISO/IEC 25010:2023 - Product quality model	Published International Standard	Vocabulary for quality characteristics and evaluation criteria	A quality model does not decide Sawbill acceptance
ISO/IEC/IEEE 29119-2:2021 - Test processes	Published International Standard	Reference for governance and organization of test processes	Sawbill retains subject, version, coverage, and evidence bindings
NIST SP 800-218 - SSDF 1.1	Final NIST publication	Secure development practices that can be integrated into the lifecycle	Following a practice alone does not prove deliverable conformance
JSON Schema Draft 2020-12	Published specification	Validation of JSON forms and contracts	Schema validity does not prove business semantics
NIST OSCAL	NIST initiative and open formats	Machine-readable exchange of control catalogs, profiles, implementations, and assessments	OSCAL transports control data; Sawbill retains applicability, evidence, and decision
NASA NPR 7150.2D	NASA directive effective from March 8, 2022 to March 8, 2027; mandatory within NASA scope	Public example of lifecycle, classification, and tailoring requirements	No non-NASA project inherits applicability through mapping alone

ISO references are cited by identifier, version, title, and public status. Their text remains subject to the publisher's rights. Sawbill mappings are neither the normative text, a certification, nor legal advice.