Scope a serious ETS discussion

Sample discovery report 02 / Independent verification gap

Fictional customer example | Authored guide, not a generated assessment or validated architecture | Version 1.0

Discovery summary

Determine how to reconstruct a consequential approval claim across three systems for an independently authorized reviewer.

Workflow: A request is approved in one system, executed in another and logged in a third.

Claim: Each deployed change corresponds to a specific authorized approval.

Current tools: Separate application logs with inconsistent identifiers and no demonstrated end-to-end export.

Impact: The organization cannot reliably reconstruct authorization for a disputed change.

Declared position: An evidence gap and independent-verification requirement are declared. Physical capture and disconnected operation are not required.

Unknowns: Minimum disclosure, identifier mapping, clock assumptions, retention, load, budget and actual development ETS capabilities remain unresolved.

What a useful result looks like

A separately authorized reviewer evaluates the selected claim from an exported package; missing or altered records produce explicit failure or uncertainty. Agree exact thresholds before the pilot. No such result is claimed here.

Read this example as a guide

Use the structure to prepare your own answers. Replace the fictional context with confirmed facts, keep unknowns explicit, and ask the proposed owners to review the recommendations. Do not copy example findings as evidence about your organization.

Actions and engagement path

1. Define one bounded claim

Select one change and identify the producer records, approval authority, linkage and permitted disclosure needed to evaluate it.

Suggested owner: Evidence owner and reviewer | Why: A precise claim makes the gap testable.

2. Establish the existing-tool baseline

Try to reconstruct that claim using current exports. Record missing links, conflicting records and unsupported assumptions.

Suggested owner: Integration owner | Why: This identifies which gap a new tool would actually need to close.

3. Design a development evaluation

Map the gap to a named development release. Require documented interfaces, limitations, evidence export and a separately authorized verification exercise.

Suggested owner: Architect and ETS engineering owner | Why: Development availability alone does not establish capability or production readiness.

ETS relevance

Potentially relevant. The declared gap and independent reviewer justify a scoped development evaluation. No ETS release or capability has been verified in this example.

Edge-device relevance

Not indicated on the supplied facts. A software-only integration should be evaluated first.

Alternatives to evaluate

Improve correlation and exports in existing systems; use a controlled manual evidence package; compare ETS only against the same claim and acceptance criteria.

Architecture discussion

Use the development ETS version to determine whether the required evidence flow is feasible. Serious tooling, integration and operational support need a separately scoped engagement and measured qualification.

Bring to a discussion

A sanitized cross-system example with identifiers / The reviewer's minimum disclosure and verification needs / A list of current export gaps and operating constraints

Scope and limits

All scenario inputs are illustrative declarations. No control, product, device or architecture has been independently verified. These examples do not assert compliance, certification, costs or quantified benefits. Manual-derived rules require approved source citations; none are asserted in these authored guides.