Audit Evidence Guide

What an assessor actually sees when your team produces calibration records from Axiospec.

This document exists so a quality manager can judge sufficiency before committing. It describes what the system produces, field by field. It does not describe what we intend to build.

Axiospec is a record-keeping system. It is not an accredited calibration laboratory and it does not perform measurements. Every artifact it generates carries that statement in print.

1. The audit pack: one click, one ZIP

Any manager, administrator, or auditor-role user can generate a complete audit archive from inside the app. It is not a support ticket, it is not a paid add-on, and it is not gated by plan tier or by being over your asset cap. Reads and exports are never blocked.

The archive is a single ZIP containing four things:

`audit_manifest.csv` One row per instrument, then one row per committed calibration event. Roughly 45 columns, including: instrument tag, name, serial number, site, location, quarantine flag, and your workspace-defined custom fields as a JSON object; then per event, the event type, workflow status, performed date, performer first and last name, reviewed date, reviewer first and last name, review notes, rejection reason, result, nominal value, tolerance, as-found reading, as-left reading, temperature, humidity, reference standard, certificate number, service provider, traceability reference, out-of-tolerance impact, measurement uncertainty, coverage factor, confidence level, decision rule, conformity statement, reference standard certificate number, and the two integrity columns `record_hash_hex` and `prev_hash_hex`.

`audit_summary.pdf` A human-readable report with five sections: a compliance dashboard (total, in-calibration, due soon, overdue, not calibrated, out for calibration, failed); a tamper-evident integrity verification block; the calibration audit trail; the master instrument roster; and out-of-tolerance detail. It ends with a "Report Reviewed and Approved By" signature line, so it can go straight into a controlled record.

`assets/<tag>/...` The actual evidence files: instrument photos and the certificates or photos attached to each calibration, grouped by instrument tag.

A verification guide (plain text, at the archive root) States the generation timestamp in UTC, the export scope, the record counts, how the hash chain works, a four-step procedure to verify the chain linkage offline, and the genesis and head record hashes for that export.

Scoping. The export can be narrowed by site, location, specific instruments, date range, and current compliance status. A scope can only narrow, never widen: it always intersects with the requesting user's own site permissions.

2. A single instrument's record

Open one instrument and an assessor sees the identity block (tag, name, manufacturer, model, serial, site, location, status, last calibration, next due) plus the full calibration history. Each history row is badged with its record state: Approved, Pending Sign-off, Rejected, Corrected, Voided, or Correction. That badge follows the record into the per-instrument history CSV, so a voided failure never exports as a plain live row.

Any single approved calibration can be printed as a Certificate of Calibration PDF. It prints: performed date, result, record status, nominal value, tolerance, as-found, as-left, reference standard, traceability reference, temperature, humidity, certificate number, service provider, calibration method, next due date, restrictions and notes, out-of-tolerance impact, measurement uncertainty, coverage factor (k), confidence level, reference standard certificate number, decision rule, conformity statement, then the attestation block (performed by, approved by, approved at UTC) and the record's own SHA-256 hash.

Two behaviors matter to an assessor:

- If your workspace has selected a target standard and that standard requires a field the record left blank, the certificate prints a visible "not recorded, required" placeholder rather than silently omitting the row. A gap is visible, not invisible.
- A voided calibration, or an original that has been superseded by an approved correction, cannot produce a clean certificate. The request returns not-found. You cannot print an official-looking certificate for a record that should not stand.

---

3. As-found, as-left, and out-of-tolerance

As-found and as-left are separate stored fields on every calibration event. They appear on the certificate, in the manifest CSV, and in the audit summary PDF.

Two separate things happen in the audit summary PDF, and they do not cover the same set of results.

The exception table. When a calibration is recorded as Fail, Damaged, or Limited Calibration, the instrument appears in the audit summary's exception table with its workflow status, result, performer, reviewer, review notes, rejection reason, and both readings. Limited Calibration is listed there for visibility, not as a failure: Axiospec treats it as a passing result carrying a caution, so do not read its presence in that table as an out-of-tolerance condition. The exception table lists the 20 most recent qualifying events; the complete set is in `audit_manifest.csv`.

The impact review checklist. The PDF then prints a separate "Out-of-Tolerance Impact Review Required" checklist, one line per affected instrument, with a Yes/No box and a blank NCR number for your quality unit to complete. That checklist covers Fail and Damaged results only. Limited Calibration does not raise a checklist line.

The impact assessment itself is a free-text field on the calibration event. It is fillable once: it can be recorded when the calibration is logged or added later through a dedicated action, and once a value is stored the database refuses to change or erase it. Assessors get a permanent record of the validity-of-previous-results decision your team made, in your team's own words.

One honest note: a failure that your team has already voided or corrected is deliberately excluded from both the exception table and the checklist. An assessor should not see a resolved failure presented as if it still stands. The superseded record itself remains in the audit trail.

---

4. Traceability to the reference standard

Traceability is captured on the calibration event itself, not on a separate spreadsheet:

- **Reference standard**, as a label, and optionally as a link to the master instrument in your own workspace that was used. That linkage gives you forward and reverse traceability inside the system.
- **Reference standard certificate number**, the certificate of the standard used.
- **Traceability reference**, the free-text chain to a national or international standard.
- **Measurement uncertainty, coverage factor (k), confidence level, decision rule, and conformity statement**, for workspaces working to ISO/IEC 17025.

Every one of these is recorded verbatim as your personnel entered it and is folded into the record's hash. Axiospec never synthesizes an uncertainty figure, a conformity verdict, or an accreditation claim. It prints what was recorded.

---

5. Corrections and voids: change without erasure

Nothing is overwritten and nothing disappears.

Corrections. A correction is a new calibration entry appended to the ledger, carrying a pointer to the entry it replaces and a mandatory written reason. The reason is rejected at the API if it is blank, and it is folded into the new entry's hash, so it cannot be edited afterward. The original entry stays in history, badged "Corrected", with all its original values readable. A technician may correct only a record they performed themselves. Managers and administrators may correct any record.

Voids. A void is a separate appended tombstone event, never a deletion. A written reason is required. Only a manager or an administrator may void, and the permission is checked both organization-wide and at the instrument's own site. The voided calibration stays in history badged "Voided", and the instrument falls back to its previous calibration, or to not-calibrated if there was none.

Why deletion is not possible. Immutability is enforced by the database itself, not by the application. A PostgreSQL trigger blocks any deletion of a committed entry, blocks reverting a committed entry to draft, and blocks changes to roughly 25 identity and measurement columns. This holds against direct SQL access, not just against the app. The only path by which committed records are ever removed is a customer-requested deletion of the entire workspace, and that happens only after a 730-day retention window during which you can still sign in and export.

Verification. Any user can run a chain check from inside the app. It walks every committed entry, confirms each record's stored hash against a recomputation, confirms each entry's link to the one before it, and names the first mismatching instrument if there is one. The result and the date are recorded. The same

check runs automatically at export time and its verdict is printed in the audit summary PDF: VERIFIED with the number of records checked, or FAILED with the record where it stopped.

An assessor can also check the chain **linkage** offline, without us. The manifest CSV carries both hash columns for every event, so every row's `prev_hash_hex` can be matched against the `record_hash_hex` of the row before it, and the bundled guide gives that procedure step by step. Any removed, inserted, or reordered record breaks that link. Be precise about the boundary: the manifest does not carry every field that feeds the hash function, so recomputing an individual record hash from the CSV alone is not possible. Recomputation happens server-side, in the app and at export time. Linkage is the part your evidence pack keeps forever, with or without the vendor.

---

6. What Axiospec does not decide

This section exists because an assessor will ask, and because a vendor that overstates here is not worth trusting.

- **Conformity.** Axiospec does not decide pass or fail, does not apply a decision rule, and does not issue a conformity statement. Those are recorded as your personnel entered them.
- **Interval acceptance.** Axiospec schedules against the interval you set and flags what is due or overdue. It does not judge whether your interval is appropriate.
- **Compliance with any standard.** Standard names shown in the product are for reference only. Conformance is determined by your auditor or certification body, not by us. We support the records; we do not certify the outcome.
- **Accreditation and measurement.** We are not an accredited calibration laboratory and we did not take the measurement. That statement is printed on every certificate, every audit summary, and in the archive guide.

---

7. Known limits, stated plainly

- **Second-person sign-off is optional.** Maker-checker review is a setting, per instrument or per workspace. When it is on, the system refuses to let the person who recorded a calibration approve it. When it is off, the submitter is recorded as the approver and the record is immediately effective.
- **No electronic signature column in the export today.** When a workspace turns on the electronic signature setting, the approver's own account password is re-verified at the moment of signing, and the mobile app requires password entry when a calibration is recorded. The archive records reviewer name, review timestamp, review notes, and rejection reason. It does not carry a separate signed-signature field. If your quality system requires a bound electronic signature artifact in the exported record, that is not shipped today.
- **Lineage is visible in the app, not as a column in the manifest.** Every void appears as its own event row and every superseded original remains present in the export. The manifest does not currently carry an explicit "superseded by" column, so the correction and void linkage is read on screen or through the API rather than from the CSV alone.

- **Instrument master-record edits are not attributed.** Calibration events are fully attributed and immutable. Changes to an instrument's own record, such as its tag, serial number, calibration interval, or location, are not currently written to a user-attributed change log.
- **A filtered export verifies differently.** A complete, unscoped export walks the entire chain end to end. A site-scoped, instrument-scoped, or date-ranged export verifies each exported record's own hash but does not assert continuity across records, because a subset is deliberately not the whole chain. The bundled guide says the same thing. For a full-integrity demonstration, export unscoped.
- **Draft entries are not chained.** Only committed entries are hashed, locked, and exported. A draft is still editable by its author until it is committed.

Questions about anything in this document, including a walkthrough of a sample archive: support@axiospec.com. Axiospec is built by CaliTech LLC in Knoxville, Tennessee.
