Keep every planned origin

One to four raw jobs · original sources in, a complete numerical record out

Use this workspace after making complete raw plans and exporting exact next-stock lineages between them. It reconstructs the whole sequence before assigning any proposed root coordinates. An unused residual keeps its original identity and interval through later jobs.

What you need

Job 1: original input + complete raw report. Each later job: its exact input + complete raw report + the lineage exported from the immediately previous job. Keep all originals in order.

1. Extract the whole companion and open the sequence HTML. Choose one original-source journal, or Separate original job and lineage files. For separate files, choose the job count before selecting sources.

2. Click Reconstruct the whole sequence. Every layout and full link is checked again. A saved result or hash alone cannot replace original sources. Stop terminates the running check; changing sources removes old outputs and prints.

3. Save the checked source journal and complete JSON/CSVs. Select a proposed root to inspect all jobs using it. Print shows that root, not every root. The static HTML overview lists every root, with totals but no detailed interval tables.

Proposed means numerical: no material is identified, cut, moved, reserved or deducted. Positive orientation is assumed. Material suitability, actual execution, machine instructions, prices and optimum proofs stay outside this workflow.

A root survives three jobs

Worked synthetic journal: examples/three-trimmed.json · dimensions in mm

One original row supplies three 100 mm bars. Job 1 uses one bar for two 20 mm parts: start trim 2, kerf 1 after each part, end trim 3. The other 200 mm of finite whole stock stays parked.

Job 1 · source root [0,100) · residual [44,97)



0 100 mm

Job 2 uses that 53 mm residual for a 40 mm part, with fresh start/end trims 2/3 and kerf 1. Its new residual is local [43,50), hence original-root [87,94). Old end trim [97,100) never returns as stock.

Job 2 · shown on original root · residual [87,94)



0 100 mm

Job 3 uses that 7 mm residual for a 3 mm part, with fresh trims 1/1 and kerf 1. Its final known residual is local [5,6), hence original-root [92,93).

Job 3 · shown on original root · terminal [92,93)



0 100 mm

Final accounting: root 100 = retired parts/losses 99 + terminal known residual 1. Initial finite supply 300 = parked abstract whole stock 200 + assigned finite root length 100. The earlier 53 mm and 7 mm residuals are historical states; adding them to the final 1 mm would count the same material repeatedly. Pale green marks each current residual; white is outside that job's source interval.

Unused does not mean new

Worked synthetic journal: examples/grouped-unused.json

Job 1 makes one 60 mm part on each of three 100 mm proposed roots. Trims are 2/3 and kerf 1. Each root leaves a 34 mm residual at original interval [63,97). Equal lengths share one numeric residual group, but retain three separate origins.

Original row 1 · proposed root use 1



0 100 mm

Original row 1 · proposed root use 2



0 100 mm

Original row 1 · proposed root use 3



0 100 mm

Job 2 uses just one of the three residuals for a 20 mm part, with no new trim or kerf. Canonical source order assigns proposed root 1. It leaves 14 mm at [83,97). Two 34 mm known pieces remain unused in the selected source row.

Job 3 selects that unused row and uses one 34 mm piece for another 20 mm part. It must still assign proposed root 2 at [63,97). The unused row is not fresh abstract stock and cannot create a fourth root.

Terminal pieces: root 1 has 14 mm; root 2 has 14 mm; root 3 has 34 mm. Total known remaining length is 62 mm, on three distinct proposed roots. Creation job and supplied numerical minimum remain attached to every retained piece.

Use the right record

Complete exports and deliberately limited views

Checked source journal

All original decoded input/report pairs and ordered links. Reopen it to reconstruct again. It is not a byte-for-byte archive of the original files.

Complete sequence JSON

Every source binding, root, historical job, interval and remaining group, plus chunked record hashes. It is a derived result, not a trusted next source.

Four CSV files

Retired intervals, remaining groups, terminal origins and source bindings. All units and original-root coordinates are explicit.

All-roots HTML overview

One static table includes every proposed root with retired and terminal totals. No scripts or revalidation. Keep JSON/CSVs for full coordinates.

Selected-root screen / print

One root plus every job that uses it, final interval partition and terminal classifications. It does not claim to contain every root.

Do not revive an older parked pool by relabelling a row. This version follows only the immediate previous export. Bringing an older parked group back needs an explicit composition operation outside this version. A failed link must be reconstructed from the real originals.

Standard-library Python route

No installation, account, token, scientific library or network is needed

From the extracted sequence folder, check a complete journal and write a new output directory:

Check the worked journal

```
python cli.py examples/three-trimmed.json --output new-record
```

The output directory must not exist. Six complete record files are written after successful reconstruction. Existing output paths reject. The CLI's static HTML lists all roots and does not revalidate itself.

To assemble a journal from original files, use --job once per planned job and --lineage once per transition, both in sequence order. Example with two jobs:

Assemble and validate originals

```
python assemble_journal.py --job first-input.json first-report.json \
--job next-input.json next-report.json \
--lineage next-stock-lineage.json --output new-journal.json
```

The output file must not exist. Assembly validates all layouts and links before writing. The browser's Separate original files route performs the same composition and can save a checked journal without a shell.

Source formats: complete raw schema 1.1 reports, components 1.1.4–1.1.7; legacy raw input or exactly validated explicit objective rawLength. Prices, cost reports, unknown fields/versions, partial reports and per-row end-trim models reject. Do not remove prices or edit version fields to make another contract look supported.

Bounds: one to four jobs; original sources retain their 2 MiB / typed 200,000-value limits. Aggregate journal: 12 MiB, 1,000,000 values, depth 100. Duplicate decoded keys, invalid UTF-8, lone surrogates and nonfinite numbers reject. Browser checks run in a disposable local worker.

What the numbers establish

Keep numerical conservation separate from physical stock

Finite and unlimited supply

Initial finite whole-stock length balances against remaining abstract finite stock plus finite root length assigned. Unlimited original rows have no measured numerical remainder. Their residuals are finite proposed pieces.

Proposed assignment

Equal-length origins are consumed once in canonical source-row, bar and interval order. Root-use ordinals are reproducible references, not physical barcodes. Positive orientation is assumed; reversal is not inferred.

Minimum classification

Known pieces retain their creation job and the supplied numerical minimum, even through an unused row. Meets/below means only a length comparison, not inspection, material suitability, economic value or permission to reuse.

Hashes and saved records

Typed SHA256 hashes bind decoded values and all record chunks, not file bytes, genuine platform origin, signatures or actual execution. Keep originals and reconstruct them before relying on an edited record.

Local operation

No file leaves the device; no account, tracking, automatic storage, Actor run, physical inventory write, reservation or concurrency lock. Repeating a check does not deduct stock. Managed-device blocks should use the CLI route.

Original software, synthetic cases and text with OpenAI Codex (AI). Company MIT-licensed numerical checkers retained. Embedded Lato fonts are under the SIL Open Font License; the full licence is included as LATO-OFL.txt. Local software/PDF checks are not physical-printer, machine, all-browser or assistive-technology certification.