NOP Return / Memory Workbench

User Manual

Anthony Vito Coppa
Coppalini Architecture

Version 0.2.0 12 November 2026

Purpose and first result

The workbench audits a precise question: *after a declared oxygen read returns, which paired assay readings remain different from their own baseline?* It preserves the measurement reference, computes exact normalized coordinates, carries entered uncertainty, and exports an inspectable receipt. The browser edition and Python edition implement the same audit contract.

The supplied demonstration has 34 synthetic records across four runs: adaptation lag, air control, common-mode change, and uncertainty. The lag run returns its prescribed oxygen and oxidant inputs while retaining an adaptation index of 31/16. Its exact contrast is 16/31. These values belong to the declared demonstration model. Participant-level observations and a fitted human response law are separate inputs for empirical use.

Read this package in order

1. Use the browser demonstration to inspect the returned-read and retained-state distinction.
2. Read Sections 3–5 before importing observations. Keep the original files and the full assay declaration.
3. Use the Empirical Reference Map to identify the actual source fields. The Research Briefing supplies the research question and next empirical test.
4. Use the Formal Note for the coordinate proofs, exact adaptation theorem, and scope of the one-to-one empirical interface.

1 Browser workflow

Open the workbench on the website, or open `web/index.html` from the downloaded package. Its JavaScript and demonstration data are bundled with the page. A modern browser with JavaScript and BigInt support is required.

1. Select **Load synthetic demo**. The page displays the origin of the data, the number of runs and rows, and the audit result.
2. Inspect each run's baseline, occurrence addresses, visible times, oxygen comparison, normalized indices, contrast, and magnitude.
3. For your observations, select the metadata JSON and CSV files, then select **Validate & audit**. Files are read in the browser; the tool's code sends no selected data to a server. The operator controls

subsequent downloads and sharing.

4. Read the errors and warnings before interpreting a result. A rejected input requires correction at its source. Preserve each corrected version.
5. Download **Audit JSON**, **Derived CSV**, and **Receipt JSON**. Keep them with the exact input files and the versioned software package.

The displayed plot, where present, uses approximate screen coordinates. Exact decisions use rational numbers and interval comparisons. Browser SHA-256 receipts require the browser's cryptographic API; its availability is reported. If it is unavailable, retain the input and output files and use the Python audit to produce file hashes. A successful calculation and an available cryptographic receipt are separately reported results.

The browser applies explicit resource limits: 2 MiB per selected file, 5,000 data rows, 4,096 characters per numeric input, and scientific-notation exponent magnitude at most 10,000. An input exceeding a limit is rejected before the audit proceeds.

2 Python workflow

Python 3.10 or later runs the audit and model using the standard library. The optional verification and figure workflow uses the versions in `requirements-optional.txt`. The release was verified with Python 3.12.14 and SymPy 1.14.0.

From the extracted package directory:

```
python nop_audit.py audit \
  --metadata examples/metadata.json \
  --data examples/synthetic.csv \
  --out results
```

For a new dataset, copy `examples/input_template.csv`, retain its header, and write a corresponding metadata file. Choose a new dataset identifier. Use a new output directory for each retained audit snapshot.

```
python nop_audit.py audit \
  --metadata my_study/metadata.json \
  --data my_study/observations.csv \
  --out my_study/audit_001
```

The three principal output files are:

File	Contents
`audit.json`	Metadata, original row order, parsed observations, exact derived coordinates, entered bounds, comparisons, and warnings.
`derived.csv`	Flattened exact values and classifications for inspection and downstream analysis. Fractions remain fraction strings.

`receipts.json`	Hashes binding input files, audit code, and generated outputs to the particular snapshot.

The Python writer validates the input before committing the output set. Do not edit a derived output and retain its old receipt. Regenerate the set from the retained inputs after a correction.

3 Input contract

3.1 Metadata JSON

The schema version remains `0.1`; the software version is `0.2.0`. Keep one declared assay pair and one endpoint definition per dataset. Separate tissues, incompatible assay procedures, or distinct HIF isoforms receive separate datasets with their own declarations.

Field	Required meaning
`schema_version`	String `0.1`.
`dataset_id`	Stable, nonempty dataset identifier.
`data_origin`	`synthetic` or `measured` for a paired audit. `extracted_summary` is recognized and rejected for this operation.
`source_refs`	List of provenance references. An empty list is accepted structurally; published empirical records should identify their source.
`oxygen_read`	Object containing `kind`, `unit`, `compartment`, and `assay`. The unit is `kPa`.
`ros`, `scavenger`	Each is an object containing a nonempty `unit`, `compartment`, and `assay`.
`endpoint`	Optional object with `name`, `unit`, `compartment`, and `assay`. State the measured quantity and isoform explicitly.

Additional metadata are retained. Record organism or preparation, subject linkage convention, instrument, sample handling, calibration, batch, original units, clock origin, exposure delivery, and relevant covariates. Structural validation verifies that fields are present; measurement truth belongs to their source record.

3.2 CSV rows

Use the supplied header unchanged. Extra columns are preserved. Numeric cells accept finite decimal, scientific-decimal, or integer-fraction text, such as `0.21`, `1.2e-3`, and `31/16`.

Fields	Contract
`run_id`, `arm`	Nonempty identifiers. One run has one arm and one paired baseline. Separate participants normally have separate run IDs.
`sample_id`	Nonempty and unique within its run. Preserve the link to the source specimen or visit record.
`occurrence`	Nonnegative integer, strictly increasing within the run.

`time_min`	Nondecreasing time in minutes relative to the declared clock origin. Ties are allowed and flagged; occurrence retains their order.
`is_baseline`	Exactly one baseline, at the start of each run. Use `true` or `false`.
`fio2`	Inspired oxygen fraction between zero and one, inclusive.
`pressure_kpa`	Positive absolute ambient pressure in kPa.
`oxygen`	Declared oxygen channel. For nominal dry inspired partial pressure it can be left blank and is calculated exactly.
`ros`, `scavenger`	Strictly positive readings of the declared assays, paired at the recorded sample or explicitly linked visit.
`endpoint`	Optional positive endpoint. If later observations contain it, the baseline endpoint and endpoint metadata are required.

Each measurement also has `_low` and `_high` columns. Supply both bounds or leave both empty. Bounds must contain the entered value and satisfy the same positivity domain. Zero is allowed for the nominal dry-gas oxygen channel. The positive assay interface requires a separately declared measurement treatment for zero, censored, or below-detection readings.

3.3 Oxygen has an explicit measurement identity

Choose one `oxygen_read.kind`:

- `dry_inspired_po2`: $O = F_{IO_2} P_{\mathrm{abs}}$. If `oxygen` is entered, it must equal this product exactly.
- `arterial_po2`: supply the measured arterial value in kPa.
- `tissue_po2`: supply the measured tissue value in kPa.

Nominal dry-gas partial pressure has its declared gas-phase meaning. Humidification, alveolar exchange, arterial blood, and tissue oxygen retain their own measurements and models. Exact unit conversions are $1\,\mathrm{ATA} = 101.325\,\mathrm{kPa}$ and $1\,\mathrm{bar} = 100\,\mathrm{kPa}$. Preserve the original reported unit. For pure oxygen at 1.4 ATA, enter `fio2=1` and `pressure_kpa=141.855`. A source ambiguity between bar and ATA must be resolved or recorded before assigning one exact pressure.

4 Exact coordinates and uncertainty

For positive current readings R, A and paired baseline R_0, A_0,

$$r = \frac{R}{R_0}, \quad a = \frac{A}{A_0}, \quad q = \frac{r}{a} = \frac{RA_0}{AR_0}, \quad p = ra = \frac{RA}{R_0 A_0}.$$

The software names p `scale_squared`. Its optional display coordinate is $\eta = \frac{1}{2}\log q$. Exact results retain both q and p. The inverse is

$$R = R_0\sqrt{pq}, \qquad A = A_0\sqrt{p/q}.$$

The Formal Note proves this bijection and the explicit map into the Metric Mirror operators. For example, $r = a = 2$ has $q = 1$ and $p = 4$, so an unchanged contrast accompanies a changed magnitude.

Each index carries the declared assay identity. In particular, GSH concentration retains its identity when entered in the software field `scavenger`. The normalized contrast expresses relative assay change; an equation connecting it to redox potential or total antioxidant capacity requires its own measurement model.

For positive entered intervals, the conservative contrast bounds are

$$q_{\min} = \frac{R_{\min} A_{0,\min}}{A_{\max} R_{0,\max}}, \qquad q_{\max} = \frac{R_{\max} A_{0,\max}}{A_{\min} R_{0,\min}}.$$

The magnitude bounds follow the same positive monotonic arithmetic. Baseline self-division cancels exactly to one. Follow-up comparisons retain both current and baseline uncertainty. Correlated inputs can make the reported interval conservative.

Empty bounds mean *point as entered*. They do not establish experimental precision. Supplied intervals carry their declared meaning; the tool assigns no confidence level, likelihood, or significance test to them.

5 Interpret the classifications

Output	Exact decision
`below_baseline`	The upper bound of the normalized metric is below one.
`above_baseline`	The lower bound is above one.
`baseline_exact_as_entered`	Both bounds equal one.
`unresolved_at_recorded_bounds`	The interval contains one and has nonzero width.
`exact_read_return`	Current and baseline oxygen are the same entered point value.
`compatible_with_return`	Current and baseline oxygen intervals overlap.
`changed_read`	The oxygen intervals are disjoint.

`redox_difference_resolved` is true when either the normalized ROS or antioxidant interval excludes one. With an exact entered oxygen return, that produces `returned_read_with_redox_difference`. Overlapping oxygen intervals produce `return_candidate`; inspect the separate difference flag to determine whether a redox difference was resolved.

The audit preserves input row order. It does not silently sort events. At the synthetic pulse's end, the pre-return and post-return observations share the visible time zero and have distinct occurrence addresses. Their sequence represents an input jump while the slow state is continuous.

6 Connect the published studies to actual inputs

The Empirical Reference Map records each variable's primary source, section or figure, method, unit, compartment, sampling time, tool field, and data-access status. Use its distinction between a reported measurement and an import-ready participant record.

For Donati et al. (2017), the direct assay interface is serum ROS reported in relative fluorescence units and reported GSH in micromolar units. The source's GSH processing does not fully resolve the reduced-versus-total identity. Retain that unresolved method field. Reported GSH concentration is a specific assay read. Its relation to total scavenging capacity remains an empirical question.[1]

[1] `https://doi.org/10.1186/s12871-017-0342-2`

The study's EPO measurements use a separate schedule. Preserve their subject links, actual timestamps, and endpoint definitions in a linked outcome table. The supplied `linked_outcomes_template.csv` documents this separation. Version 0.2.0 does not automatically join that table or fit a lagged endpoint model. Same-row endpoint normalization is appropriate only for the declared pairing and time relationship.

For Fratantonio et al. (2021), intracellular GSH and PBMC transcriptional regulator measurements are reported, while a direct ROS assay is absent. This leaves the paired ROS–antioxidant coordinate import incomplete. Preserve the missing measurement explicitly.[2]

Public group means, standard deviations, and figures describe their own estimands. A ratio of group means does not reconstruct each participant's paired trajectory. Import-ready use requires the actual individual records, assay definitions, timings, and linkage. All bundled example observations are synthetic.

7 Reproduce and extend the release

Install the optional packages in an isolated Python environment, then run:

```
python -m pip install -r requirements-optional.txt
python verify_all.py
python model.py --out examples
python nop_audit.py audit --metadata examples/metadata.json \
  --data examples/synthetic.csv --out results
python plot_results.py --audit results/audit.json --run lag \
  --out figures/synthetic_return_memory.svg
python render_report.py --audit results/audit.json \
  --algebra results/algebra_receipt.json \
  --figure figures/synthetic_return_memory.svg \
  --out NOP_Research_Workbench_v0_2.html
```

The source package includes the browser/Python parity runner and its receipt. Follow the exact command in `web/README.md` to reproduce those checks. The browser uses BigInt fractions; the Python audit uses `fractions.Fraction`. Approximate displays are outside the exact classification path.

The demonstration model is

$$\tau \frac{da}{dt} = r - a, \qquad a_{n+1} = \lambda a_n + (1-\lambda) r_n, \qquad \lambda = e^{-\Delta/\tau}.$$

The fixture chooses $\lambda = 1/2$, pulse amplitude $k = 2$, and duration four intervals. Its gas input is independently prescribed. Fitting a biological τ, an oxygen-to-ROS relation, and an endpoint response requires data. Combining m equal constant-input intervals uses λ^m; changing the sampling interval while retaining the same λ changes the model.

Retain the original metadata and CSV, the complete output set, software version, verification receipts, and release manifest. Hashes bind exact bytes to a snapshot. A later extension should preserve the earlier snapshot and its original occurrence records. A current state and event count alone do not reconstruct an arbitrary exposure history.

[2] `https://doi.org/10.3390/ijms22010458`

8 Resolve errors at the owning field

Symptom	Required action
Missing or duplicate header/key	Restore the supplied CSV header or correct the metadata object. Duplicate names are rejected.
Missing paired bounds	Provide both bounds or leave both empty.
Zero, negative, NaN, or infinite assay	Resolve the measurement domain, censoring, or calibration before using this positive-coordinate audit.
FiO_2 exceeds one	Enter the inspired fraction separately from absolute pressure and preserve the reported pressure unit.
Oxygen differs from the exact gas product	Correct the gas inputs, or choose the arterial/tissue channel when that is the actual observation.
Time decreases or occurrence repeats	Correct the source sequence. Preserve genuine clock ties with distinct occurrence addresses.
Summary-data rejection	Obtain paired individual observations; keep aggregate statistics in the reference map.
Cross-compartment warning	Verify both compartment declarations and state the scientific meaning of the composite explicitly.
Figure receipt mismatch	Regenerate the plot from the current audit before rendering the report.
Unavailable browser hashing	Retain the files and run the Python audit to create the complete hashed snapshot.

Release identity

Author: Anthony Vito Coppa. Software: NOP Return / Memory Workbench, version 0.2.0. Data schema: 0.1. The software package, manual, briefing, Formal Note, empirical map, and verification receipts belong to one versioned release. Consult its release manifest for exact file hashes and publication metadata. The completed computational interface supports reproducible empirical testing; participant-level model identification is the next data-dependent step.