

Make Any State — CQE Workbook (v1)

Generated: 2025-09-20T18:45:38.807291Z

Purpose: Teach you to synthesize any candidate state and run the CQE loop.

Make Any State — CQE Workbook (v1)

****Purpose:**** Teach you to synthesize *any* candidate state (a d10 torus with double-helix rest) from scratch, bind its meaning via glyph labels, and run it through the CQE loop.

Part A — Name and Lock Meaning (Glyph Hash)

1. Pick a glyph string that *already* reflects the source data's public semantics*.

Examples: `□◇◆♣A`, `□∞◇R`.

2. Write it in `label_glyphs`. This is a *meaning-hash token*. It must not change during a session.

> Tip: Use domain-native glyphs (bio, em, thermal) so the hash also serves as a self-constraint.

Part B — Choose Form and Pose

- `form_tag`: `torus_d10_doublehelix` (default), or `leech_patch`, `e8_octad`, etc.

- `pose`: `rest` (start), then try `tilt`, `braid`, `weave` if mirror fails.

- `parity_lane`: 0 or 1 (pick; its mirror must exist).

Part C — DNA-10 (state save)

Fill `dna10_template.csv`:

- timing = `13xN` (or your cadence)

- polarity = `mirror_on`

- scale = `1,2,4,8,16,32,64` (choose)

- pose = from Part B

- domain = coarse label (EM, BIO, COSMOS, ...)

- conditioning = `well-posed` initially

- units = `SI`

- precision = `1e-3` to start

- cost = note
- seed = any int

Part D — Octet Plan

Open `octet_plan_template.csv` and define 8 materially independent views.

Examples (adapt to your domain):

1. Airy → PSF
2. FFT pupil
3. Zernike sweep
4. Phase retrieval
5. Radiator $\sigma \epsilon AT^4$
6. Opposed vents cadence
7. Polar L/R & Linear X/Y
8. Units guard / interval bound

Part E — Mirror Test

Fill `mirror_test_template.csv` with forward ↔ inverse pairs and a tolerance.

Run each pair; record `passes` as `TRUE/FALSE`.

- If any fail, perform a **Δ-lift**: a *local* repair (e.g., adjust a bias, narrow a vent, tweak a stem length).
- Re-run mirror; repeat until stable.

Part F — Strict Ratchet

Once the mirror is stable, **tighten** one bound (e.g., tolerance $1e-3 \rightarrow 8e-4$).

Re-run octet + mirror; only keep the tightened bound if all still pass.

Part G — Receipts & 4-bit

Record:

- OPE/FCE debt
- mirror_votes, view_votes
- compute `fourbit` from octet bits (see harness)
- store `hash` (mirror receipt)

Commit status:

- `working` if ≥ 4 octet views pass and mirror is stable
- else `provisional` or `non-working` (but keep breadcrumbs)

Part H — Export the State

Append a new row to `datasets/cqe_demo_dataset.csv` with fields:

`state_id, label_glyphs, form_tag, parity_lane, scale, pose, domain_hint, sidecars_active, octet_passes, mirror_hash, fourbit_commit, residues, status, notes`

****You're done.**** This state is now replayable and comparable.

Appendix — Quick Recipes

****Δ-lift cookbook****

- Optics: remove astig (Z2), refit with lower WFE bound.
- Thermal: shorten vent window, raise emissivity model to conservative prior.
- Polar: tune LC bias by -0.1 V to recover axial ratio.
- MATH: rewrite mixed units to SI; re-evaluate bound.

****Strict ratchet rules****

- Tighten one thing at a time; only after a pass.
- Never loosen a bound silently—log it or annihilate the path.

****Viewer grids****

- Local 8×8 for each octet view; global $4 \times 4 \times 4 \times 4$, and parity halo $2 \times 2 \times 2 \times 2$.