

Refresh and Recertification — How the Curriculum Stays Current

The doctrine is durable; the canon is not. The FAR is amended, Shipley publishes new editions, the DNA corpus ships new versions, and every debrief teaches a lesson the previous curriculum did not contain. This module is the discipline’s own learn stage, applied to the discipline itself.

Level: instructor + forever-learning · **Companions:** course/forever-learning-program.md, modules/forever-learning/README.md, align/concept-to-system-map.md.

This module is the **refresh machinery** that keeps the forever-learning promise real. It answers one question: *how does a curriculum that refuses to rewrite its doctrine stay current for twenty years?* The answer has three parts — the two-layer rule that protects the doctrine, the refresh cadence that tracks everything else, and the recertification that keeps every learner and instructor re-grounded on that cadence.

1. The refresh problem

The curriculum lives on a durable core and a changing surface:

Layer	Durable or changing	What happens to it
The doctrine (doctrine/01–10)	Durable	The concepts — money, pipeline, gates, ORBITAL, lifecycle — do not change. They are not rewritten to chase the stack (doctrine/08).
The canon (literacy/)	Changing	The FAR/DFARS parts, the APMP BOK, Shipley’s editions, the certification landscape. These must be tracked as they change.
The machine corpus (DNA)	Versioned	Rules, formulas, playbooks, prompts ship under version numbers. The human rendering must stay in lockstep with them — “one doctrine, two renderings.”

Layer	Durable or changing	What happens to it
The outcome record	Ever-growing	Every debrief, protest, and award adds a data point. Lessons learned must be retrofitted into the tiers where they belong.
The modules and courses	Refreshed	Teaching plans are re-checked against the doctrine and the canon; examples and cases are updated; the tiers are re-rendered.

The discipline’s mistake would be to treat the whole curriculum as durable, or the whole thing as changeable. The first makes the curriculum go stale; the second makes it unteachable. The refresh machinery exists to get the **layering right**: the doctrine stays still, and everything else is kept honest against it.

2. The two-layer rule protects the doctrine

The refresh cadence is only safe because the curriculum enforces the two-layer rule from the repo’s own operating instructions (CLAUDE.md):

- **The doctrine never changes to chase the stack.** When a portal, an AI capability, or a product changes, the doctrine is not rewritten. The change belongs in the alignment map (`align/concept-to-system-map.md`), which maps each durable concept to its current machine-side encoding — the *stack column* changes, the *concept column* never does.
- **The canon layer is where regulation lives.** Government facts and the rulebook’s current state live in the `literacy/` layer, which is deliberately the layer that *does* track the FAR, the APMP BOK, and the certification landscape. When a fact changes, the literacy layer is updated — not the doctrine.
- **The renderings stay in lockstep.** Every DNA version bump (the machine side) triggers a re-sync check on the human side: do the concept-level lessons still match what the machine encodes? The refresh cadence is what makes “one doctrine, two renderings” a *maintained* claim rather than a hope.

3. The refresh cadence

The cadence is three engines running on different clocks:

Engine 1 — Annual recertification (the year clock)

Every learner and every instructor **re-grounds once a year**. Recertification is a re-reading, not a re-taking — it is not a punishment and not a gate that fails. It is the discipline’s equivalent of a pilot’s currency check: a structured re-acquaintance with what changed and a re-verification that the durable core still organizes it.

The annual recertification has three standing deltas:

1. **The rulebook delta.** What changed in the FAR, DFARS, DCAA posture, CAS coverage, and TINA thresholds over the past year — read as a strategic signal, not a compliance briefing.
2. **The canon delta.** What Shipley, the APMP, and the certification landscape published or revised — read so the professional’s vocabulary stays current.
3. **The practice delta.** What the outcome record and the debrief-retrofits taught over the past year — read so the professional’s judgment absorbs the field’s new lessons.

For an instructor or program owner, recertification adds a fourth delta: **the curriculum delta** — what was retrofitted, re-rendered, or re-versioned, and whether the courses they teach reflect it.

Engine 2 — DNA-version-aware updates (the version clock)

The machine corpus ships under version numbers. Every version bump triggers a **human-side re-sync**: a comparison of the machine encoding against the human rendering, concept by concept, using the alignment map as the key.

The re-sync answers three questions per concept:

- **Did the concept change?** It should not — if a version bump silently alters a durable concept, that is a defect to escalate, not an update to absorb.
- **Did the encoding improve?** If a rule, formula, or playbook got sharper, the human rendering should reflect the improved *concept-level lesson* where the course teaches it.
- **Does the map still point?** The `concept_system_map` in the manifest and the alignment map’s stack column must be updated so the next reader finds the current artifact.

The DNA-version-aware update is the mechanism that keeps the human curriculum from drifting from the machine doctrine — the same drift the wider ecosystem treats as a defect (the dashboard’s standing order: never let it drift).

Engine 3 — Debrief-retrofit (the event clock)

The most important source of new lessons is the **outcome record** — the debriefs, protests, and lessons-learned reviews the field produces continuously. The

```

graph TD
    A[a real loss] --> B[the debrief mined for the lesson]
    B --> C{does the doctrine already teach this?}
    C -- yes --> D[the lesson is confirmation, not retrofit]
    C -- no or not sharply --> E[the lesson drafted as a written, reviewed retrofit]
    E --> F[placed in the tier where it belongs]
    F --> G[re-rendered]

```

4. Recertification for learners and instructors

Member	What recertification is
Practicing professional	A structured re-grounding on the three deltas, plus a re-read of one doctrine chapter at altitude (the one most relevant to their year's work).
Alumni mentor	The three deltas plus a re-verification of the practicum's current teaching objects (the case library's new entries).
Instructor / program owner	The four deltas (add the curriculum delta) plus a review of every module they teach against the current canon and doctrine.
Post-doctoral fellow	The research-agenda delta — what the field learned and what is now the open question — plus the corpus curation backlog.
Executive	The regulatory-change watch's year in review (executive-education.md), read at strategic altitude.

4

status: the learner and the instructor have re-grounded, the deltas are absorbed, and the discipline is trusted to keep teaching.

5. Ownership and honesty

The refresh machinery has named owners, because a cadence without an owner is a wish:

- **The curriculum delta** is owned by the program’s editorial board (the instructor side of the program overview).
- **The DNA re-sync** is owned jointly with the machine side — the post-doctoral fellowship’s curation role is the human-side owner of the lockstep.
- **The debrief-retrofit pipeline** is owned by the community: study groups and executive roundtables produce the candidate lessons, the symposium is the gate, and the editorial board ships them.
- **The literacy layer** is owned by the program’s fact-checkers, under the literacy layer’s accuracy rule (literacy facts must be accurate; if not certain, state it generally).

And the whole machinery is governed by the curriculum’s honesty standards: no invented specifics, no fabricated evidence, no misattribution to real organizations. A retrofit is only as good as the debrief it came from, and a version bump is only as good as the integrity of the corpus it ships in.

Self-check

1. Why is the refresh cadence *only safe* because of the two-layer rule — what would go wrong if the doctrine were treated as changeable?
2. What are the three clocks of the refresh cadence, and which one does a debrief fire?
3. What is the difference between a DNA-version *update* (which is absorbed) and a DNA-version change to a *durable concept* (which is a defect)?