Literacy — How the Machine Learns

The whole doctrine you are learning also has a machine rendering — a corpus of rules, playbooks, formulas, prompts, and retrieved knowledge that an AI agent executes to run the same pursuit pipeline you study here. This page opens that door. It is the surface layer between the human curriculum and the machine-side doctrine: by the end, you should realize the doctrine is also executable by machines — and that you still own the decisions.

This is a concept-first page, like everything in this curriculum (Doctrine 08): it describes **what the machine holds and how you judge it**, in durable terms, with no product knowledge required. There is no tool to install, no system to operate. What there is, is a way of thinking about the machine half of your profession that will let you read its output, catch its errors, and keep the human accountable.

The same doctrine, rendered for machines

You are learning the doctrine as prose — chapters, gates, self-checks. The machine side renders the **same doctrine** as executable knowledge: a corpus an AI agent reads and runs. Nothing about the concepts changes between the two renderings. What differs is the *form*:

What the corpus holds	What it is, in durable terms	You already know it as
Rules	Deterministic, if-then decision logic — the gates, the knockouts, the floors, the compliance checks	Doctrine 04 (gates), Doctrine 02 (compliance), Doctrine 05 (pricing floors)
Playbooks	Step-by-step procedures for a whole discipline — capture, proposal, post-submission	Doctrine 03 (the pipeline), Doctrine 09 (capture)
Formulas	Arithmetic that turns inputs into a number — the score, the price-to-win position, the readiness index	Doctrine 05 (score = pWin × value; price-to-win)
Prompts	The standing instructions that make the agent *play a role* — its persona, its stance, what it checks	Doctrine 07 (the lifecycle's stages)

What the corpus holds	What it is, in durable terms	You already know it as
Retrieval corpus	A library of reference knowledge the agent draws on — past performance, winning themes, agency patterns	The literacy layer of this curriculum

The name you will hear for this corpus is **the machine rendering** (or "the machine-side doctrine"). The exact tool that runs it is a swappable detail — what matters is that the concepts it executes are the ones this curriculum teaches, which is exactly why the concepts must be durable and the tools swappable (Doctrine 08).

Who the machine thinks you are: the skill personas

The machine rendering works by playing the **roles a pursuit organization staffs** — the same roles Doctrine 09's staffing model names. Each persona is a standing set of instructions that tells the agent what to do, check, and produce in that seat:

Persona	What it produces for the pursuit
Capture manager	The win strategy, the customer picture, the capture plan, the go/no-go position
Proposal manager	The production plan — schedule, page budget, compliance matrix, review cadence
Price-to-win analyst	The price position — cost floor, competitive range, the price the story supports
Bid/no-bid analyst	The scored recommendation and the disposition: bid, bid with conditions, or no bid
Compliance officer	The compliance matrix and the eligibility screen — every "shall," every certification
Discriminator analyst	The three-to-five discriminators that move the score, tested against the competition
Past-performance narrator	The narratives that turn delivered work into evidence (the CPAR-backed Volume III)

Persona	What it produces for the pursuit
Storyboard author	The section-by-section promise: what each page will prove and with what evidence
TRL assessor	The technical-maturity read for R&D and SBIR/STTR pursuits
Critic roles	The color teams — pink (compliance), red (persuasiveness), gold (executive decision), black-hat (the adversarial read), white-glove (production)

When you read a machine output, ask which persona produced it — because each persona's output is checked against that persona's standard. A past-performance narrative is judged by whether it evidences delivered work; a discriminator claim is judged by whether it survives the black-hat read.

The rules of the road: eligibility, security, and the boundary

The machine rendering does not just pursue — it *polices* the pursuit against the rule layer you study in Doctrine 01, 02, and 04. Three families of controls matter most:

- **Eligibility.** The machine applies the **SBA size standard** and the **NAICS alignment** to the pursuit before anything else — a hard knockout, the same way Doctrine 09 teaches qualification. If the size or the set-aside does not fit, the machine says no, and it should say no loudly.
- **Security.** The machine checks the solicitation against a security-control catalog (the **NIST 800-53** control set is the one federal programs cite) — what data is touched, what system boundary applies, what the offeror must certify. Security posture is a compliance factor, not an afterthought.
- **The integrity boundary. Data rights** (who owns what intellectual property and under what license) and **organizational conflict of interest (OCI)** (whether the offeror has a position that makes an unbiased award impossible) are the walls the machine must not cross — and must flag when a human-written approach is about to cross them. The machine that is silent on an OCI is failing at its job.

None of these are "the machine's own rules." They are the public rulebook, encoded — and you, the professional, are responsible for knowing them well enough to catch the machine when it misapplies them.

The analytical machinery

Beyond the deterministic rules, the machine rendering carries **analytical machinery** — the arithmetic and scoring models that turn judgment into numbers.

You already hold most of these as concepts from Doctrine 05 and Doctrine 09; here is how they appear on the machine side:

- **Calibration** — the discipline that a probability estimate must be *true*: an agent that says 0.7 should win about seven times out of ten. The machine watches its own calibration against real outcomes and is supposed to reweight when it drifts. A pWin estimate that is never checked against reality is a guess wearing a number's clothes.
- **TRL** — the **Technology Readiness Level** scale that grades technical maturity, used on research and SBIR/STTR pursuits. The machine scores the proposed technology's readiness so the evaluator can see the risk.
- **RICE** — a prioritization scorecard for *which pursuit to work on*: **Reach** (how much it matters) × **Impact** (how much a win changes things) × **Confidence** (how sure we are) ÷ **Effort** (what it costs). It is the arithmetic of finite capture capacity — the same capacity Doctrine 04's gates protect.
- **Gap-point-delta** — the machine's measure of *how far the pursuit stands from where it must be*: the delta between the current position and the requirement at each gate. It is the arithmetic version of the readiness review — and when the delta is too wide, the answer is defer or withdraw, not "start writing anyway."
- **Incumbency pressure** — a score of how much pressure the incumbent is under (contract ending, churn signals, customer dissatisfaction in the public record). It feeds the displacement strategy Doctrine 09 teaches — the opening where a challenger wins.
- **Strategic positioning** — a composite of the pursuit's competitive strength (discriminators × evidence × fit) at the readiness gate. It is the machine's answer to "are we actually positioned to win?"
- **Agency win patterns** — the machine's harvested patterns of what reliably wins at each agency and funder family: which narrative structures, which evidence, which pricing postures. Like a competitive-intelligence analyst (GE 630) who never sleeps.
- **BOE / cost-volume construction** — the machine assembles a **basis of estimate**: every cost element, its quantity, its unit cost, and its source, then renders the cost volume — the technical proposal in money. A BOE with no source is a guess, whether a human or a machine wrote it.

How a professional reads machine output (D8.5)

The single most valuable skill this page gives you is **evaluating the machine's work** — competency D8.5 in the model: *is it right, not just plausible.* A fluent, confident machine output is not a correct one. The discipline is:

- **Check it against the doctrine.** Every machine output is a claim about a pursuit. Run it against the concepts you know: does the discriminator survive the black-hat read? Does the BOE have a source per element? Does the compliance matrix cover every "shall"? A machine output that violates the doctrine is wrong even if it is perfectly formatted.

- **Know when to trust.** Trust the deterministic, the verifiable, and the well-scoped: compliance sweeps, format checks, arithmetic, retrieval of a known fact from the corpus. These are where the machine is reliably better than a tired human at 2 a.m.
- **Know when to override.** Override on judgment, novelty, and the irreversible: a new competitive situation the corpus has not seen, a pricing call where the floor is a business decision not a formula, a narrative that needs human credibility behind it, and anything touching ethics, OCI, or the integrity boundary. The machine proposes; you dispose.
- **Keep the human in the loop at the Dream Gate.** The gates that commit real resources — bid/no-bid, readiness, the submission decision, the financial commitments — keep a human Accountable. The machine can draft, score, and warn; it does not sign. The whole point of the gate discipline (Doctrine 04) is that the criteria are written in advance and the *owner* makes the call. The owner is a person.

The seam

This page is the surface layer; the actual correspondence between each durable concept and its machine encoding lives in the **alignment map**: `align/concept-to-system-map.md`. That is the seam between the two renderings — one table per concept, with the concept column durable and the machine/stack column swappable. When you want to see exactly which rule, formula, or persona encodes a given chapter, that map is where you look.

One practice loop

1. Take a doctrine chapter you have read — say, Doctrine 05 (scoring and price-to-win).
2. Ask: *what is the machine rendering of this concept, and which persona or formula executes it?* (Check the alignment map to confirm.)
3. Then take a machine output you have seen — a score, a compliance sweep, a narrative — and run D8.5 on it: *is it right, or just plausible? What would I override, and what would I trust?*

You now know what the machine half of this discipline actually is: the same doctrine, rendered as executable knowledge. It can run a pursuit, but it answers to you — and the day you stop checking its work is the day you stop being a professional.