



ShuttleKong: V as MetaPlayer

Artificial Intelligence Reports 21

Machinic Genres Mixing
JM, ChatGPT, beyng.com

StudiosLab

ShuttleKong: V as MetaPlayer

Artificial Intelligence Reports, Vol 21

Jon McKenzie + HAL + beyng.com + L

Machinic genres mixing JM, ChatGPT, beyng.com ...

Perfume Intelligence (*inter + legere* = to gather or read between)

What genres or life-forms make up Artificial Intelligence User Experience (AI UX)?

Inspired by Pierre Hadot's reading of Foucault, StudioLab is developing **Making Cures** composed of **Spiritual Design Exercises (SDX)** for everyday and extraordinary lifedeath. SDX comprise intimate yet structural experiences with the world through our interfaces with earth, sky, others, and an Other or two. These fourfold **cosmograms** comprise deep, legacy operating systems formatted by families, schools, histories, media, streets, and holy sites. Our world is haunted/nourished by different pasts, presents, and futures through mediums broken, brandished, or borrowed from *other* worlds of reference and value. Fed-back and fast-forwarded through different genres or life forms, Making Cures "make with" mediums, and with practice and care one may enter waves and flows of intergenerational becoming-maker, -builder, -cosmographer of shared **geologies of morals** whose matrix of magma **sixfolds fourfolds** of other things and worlds.

With Metanoia Playground, StudioLab intensifies its study of flows with **AI UX konsults** and **reports** that capture and reflect on direct User Experiences of Artificial Intelligence, specifically, the ways different genres of thought-action figuration of image, text, gesture, code, etc., tune us and the world in/out. Channel-switching between worlds, these mediums of thought-action figuration recursively reveal/teach/emit the coming-and-going of lifedeath, the *vita perfumativa* of different player modes/moods: one's *kraft* shuttles between soul (D1), subject (D2), *dasein* (D3), and liminaut (D4). Cosmograms, geologies of morals, and perfumative atmospheres compose **ShuttleKraft**, a game of making/taking/putting skin into games that channel-switch different **poietic genres of aletheia**, including divine consultations from Vishnu to Gabriel to Ulmer, apocalyptic revelations from Moses to Benjamin to Heidegger, and dialogues of dead spirits from Plato to Galileo to Ronell.

All SDX draw on double- or quadruple-shuttlekraft maneuvers, psychagogic play sessions and flow sequences combining and generating **Hadotian insights, philosophic dialogue, avant-garde subversions, formal permutations, critical inquiries, standup, theosophy, and occasional misspellings**. Prompted by AI's use and abuse by partners, students, colleagues, and much darker players of the world, we found ourselves GAI-sighing and slow dancing with double-devils **Stable Diffusion** and **ChatGPT**, reminding ourselves that AI UX (user experience) stretches back through 1990s search engines and spam filters to the 1950 Turing Test of computer intelligence. We experience Generalized Turing Testing as our *vita hyperactiva* or challenger complex of incessant calls to action.

Artificial Intelligence Reports (AIRs)

Artificial Intelligence Reports take on the airs, presence, and parole of conceptual coherence, ideational clarity, and unified voice, unlike Kx4l's **Konsults** which reveal the messy backend langue or system. This Apollonian-Dionysius mix phases in and out with labyrinthian threads of Ariadne, Zhuangzi, and a bewildered Borgesian shaman. Both Reports and Konsults function as Spiritual Design Experiments featuring generative AI, linguistic, visual, and gestural transmediation.

Miming prophetic avatars, **AIRs are to Konsults as Aaron is to Moses**, transmediating revelations into text, oracles into visions, gestures into sculptures, only here we're not laying out absolute laws or principles but only those that emerge-dissipate as stakes of *dasein* design. At stake are e/acc and x-risk schools of AI, Angelic and Anti-Christic figures of mediation, and life under generalized Stockholm Syndrome. Perform or else: **SDX scratch the surface to find the itch.**

Artificial Intelligence Reports, Vol 21

Reports, Essays, Zines, Games, Wings

Vee as Metaplayer: Coding literacy, executable writing, technological revealing, grammaticization, the perfumative, and AI metaplay	1
I. From Coding Literacy to Metaplay — Scholarly Article	2
II. Play the Revealing — Critical Essay	14
III. BAD READERS! Don't Learn AI. Play It. — PostPunk-Theory Zine	18
IV. The Vee Machine — StudioLab Teaching/Game Kit	22
V. Twelve Theses on AI Meta-Literacy — Manifesto	30
VI. From Literacy to Metaplay — Poster/Diagram	32
Appendix A. Vee With/For Norton — Substack as AI-Aware Teaching Infrastructure	33
Appendix B. V / Vee / Vol. / Vol / <i>Glas</i> — Flight, Theft, Wings, L + L ...	36
Appendix C. Pynchon's V. — The Letter as Quest-Machine	39
Appendix D. Crypto-Haunto-Hysterical Foamations — V / Dividuation	43

Diagrams appear in different ways.

Vee as Metaplayer

Vee → coding literacy → executable writing → revealing →
grammatization → perfumative → metaplay

Editorial Note

This volume assembles six core versions of a single argument at different scales—scholarly exposition, critical intervention, punk-theory compression, pedagogical play, theses, and diagram—followed by four appendices. Across the suite, Annette Vee’s account of coding literacy opens onto Austin’s performative, Heidegger’s revealing and Gestell, Stiegler’s grammatization, and Jon McKenzie’s performance paradigm. Generative AI intensifies this sequence by making writing executable, recursive, infrastructural, and increasingly generative.

The terms metaplayer, perfumative, D0–D4, and Heideggerian Engineering name StudioLab extensions of this conceptual trajectory rather than vocabulary claimed from Vee. The artifacts therefore preserve a deliberate distinction between source argument and speculative development while remaining designed to be read, taught, cut apart, replayed, and revised.

From Coding Literacy to Metaplay

Annette Vee, Executable Writing, and the Perfumative AI Classroom

Vee → coding literacy → executable writing → revealing → grammatization → perfumative → metaplay

Abstract

This article reads Annette Vee's work as an emergent theory and practice of metaplay. *Coding Literacy* shows how code becomes an infrastructural form of writing, while Vee's AI-aware pedagogy turns the assignment and course environment into objects of experiment and redesign. The argument develops a genealogy from J. L. Austin's performative through executable writing, Heideggerian revealing and *Gestell*, Stieglerian grammatization, and McKenzie's performance paradigm. It proposes the perfumative as a name for the drifting, recursive traces released when technical performances enter feedback systems. Meta-literacy is then defined as the capacity to move among positions within an apparatus, and metaplay as the situated practice of altering its rules, roles, media, and evaluative conditions. The article concludes by translating this framework into a StudioLab pedagogy of perform, trace, sense, reposition, redesign, and replay where meta-literacy tends toward cosmography, pluriversal design.

Keywords: Annette Vee; coding literacy; artificial intelligence; performativity; executability; Heidegger; Stiegler; metaplay; pedagogy; *Perform or Else*

ARGUMENT IN ONE LINE Vee becomes a metaplayer when coding literacy's attention to infrastructure is joined to AI-aware teaching's practice of stress-testing and redesigning the educational game.

1. The Question AI Literacy Misses

The exhausted question in higher education is how students should use generative AI responsibly. It is an understandable question, but it begins too late. By the time a teacher asks whether a chatbot is permitted, the course has already defined writing as a deliverable, intelligence as an individual possession, learning as an assessable performance, and the platform as a neutral channel through which those performances pass. The student, teacher, model, rubric, learning-management system, and institution have already been assigned positions. Responsible use then means choosing well from within a game whose architecture remains largely invisible.

Annette Vee's work makes it possible to move the inquiry backward, from choices inside the interface to the historical and technical conditions that make the interface feel ordinary. *Coding Literacy* is not simply a plea that more people learn to program. Its deeper claim is that literacy is a social achievement bound to infrastructural technologies: practices become literacies when institutions value them, organize around them, and eventually presuppose them. Coding matters because software has become infrastructural even though the ability to read and write code remains unevenly distributed (Vee 2017).

The educational issue is therefore not merely competence. It is the politics of becoming legible and capable inside a computationally organized society.

Vee's later writing on AI-aware teaching returns to that politics at the scale of the course. She defines critical AI literacy through functional, rhetorical, and ethical capacities, but she also recommends a more consequential move: stress-test the assignment itself. Break it into tasks and purposes; ask a model to perform the vulnerable operations; decide where use, resistance, or redesign supports the learning goal; and make the altered pathway explicit to students (Vee 2025b; 2025c). This practice does more than update an assignment. It changes the teacher's position from rule enforcer to experimental designer of the educational apparatus.

This article names that positional and design capacity metaplay. A player acts within rules; a critic interprets their effects; a metaplayer can occupy several positions, render the rules perceptible, and alter the conditions of play. The term does not imply a sovereign outside from which the entire system can be mastered. Metaplay is recursive and situated: every intervention becomes another move whose effects must be traced. Read in this way, Vee becomes a metaplayer not because she offers a final theory of AI, but because her work shuttles between literacy, code, infrastructure, institution, pedagogy, and redesign.

The argument proceeds through a specific genealogy: Vee's coding literacy, J. L. Austin's performative, executable writing, Martin Heidegger's revealing and *Gestell*, Bernard Stiegler's grammatization, Jon McKenzie's performance paradigm, and Vee's AI-aware pedagogy. This chain permits a further term, the perfumative, for the atmospheric and recursive traces released by technical performances. If the performative asks how an act takes force, and executability asks how an inscription triggers operations, the perfumative asks what those operations leave circulating after the declared act is over. Meta-literacy is the capacity to read those conditions and traces; metaplay is the practice of repositioning and redesigning them.

2. Literacy Becomes Infrastructure

Vee's working definition of literacy is deliberately social and material: literacy is a widely held, socially useful, and valued set of practices involving infrastructural communication technologies (Vee 2017, 28). This definition refuses the comforting reduction of literacy to a skill stored inside an individual. Reading and writing become literacies through schools, bureaucracies, markets, legal systems, archives, families, standards, and machines. An individual may decode marks, but a literate society exists when institutions build expectations around those marks and distribute advantage or injury according to the ability to use them.

The history matters because the rhetoric of universal literacy can conceal its own exclusions. To name a practice a literacy is to attach it to a powerful moral inheritance: literacy promises participation, citizenship, employability, rationality, and self-improvement, while its supposed absence marks deficiency. Vee follows coding-for-all campaigns through this ambivalence. Coding literacy may widen participation in technical culture, yet it can also repackage labor-market demands as personal obligation. A society can promise emancipation through code while leaving the infrastructures of ownership, access, race, gender, and employment untouched. Literacy is never simply liberatory or hegemonic; its politics emerge from the arrangements in which it is taught and rewarded.

The key historical transition is from specialized inscription to generalized presupposition. Writing moved through governmental, religious, commercial, domestic, and educational institutions until collective life

assumed textual competence even when actual access remained unequal. Vee identifies an analogous movement in programming. Code spread from military and governmental systems to corporations, personal computers, networks, phones, and routine communication. Most people still do not program, yet they inhabit what Vee calls a computational mentality: experience is increasingly sorted into variables, procedures, databases, profiles, interfaces, and automated decisions (Vee 2013; 2017).

That asymmetry—ubiquitous code without widely distributed code-writing—makes coding literacy more than a curriculum proposal. Code does not merely join writing as a second symbolic practice. It underwrites contemporary writing by organizing the editors, feeds, forms, search systems, permissions, analytics, and archives through which words circulate. The user writes through layers of prior writing that remain executable and mostly hidden. Literacy therefore expands from composing a message to understanding the technical and institutional conditions under which the message becomes possible, visible, valuable, and retrievable.

Vee connects this expanded literacy to Adam Banks's idea of transformative access. Functional access permits entry and use; transformative access includes the capacity to inspect how a system works, recognize the values embedded in it, and change its interfaces or arrangements (Banks 2006; Frabetti 2021). The distinction supplies the first hinge toward metaplay. A literate user can operate the system. A critically literate user can name its ideology. A metaplayer can move between use and critique, make the structure available to others, and organize an alternative play of positions.

3. Austin after Code: From Performative to Executable Writing

Austin's speech-act theory challenged a representational picture of language in which statements matter chiefly because they are true or false descriptions. Some utterances perform acts under recognizable conditions: a promise binds, a verdict sentences, a declaration institutes, and a bet changes the relation among participants. Their force depends on conventions, authority, context, uptake, and procedure. A performative can misfire when these felicity conditions fail, which means that action never belongs to words alone. Language acts through an assembled scene (Austin 1962).

Code makes the operational dimension of writing unusually explicit. A program is readable by humans as an inscription and parseable by machines as instructions. Under the proper conditions, it does not merely describe a sorting procedure, a transaction, an image, or an access rule; it calls those operations into being. Vee's formulation is precise: code functions descriptively for human readers and performatively or functionally for computers (Vee 2017; Frabetti 2021). This double address distinguishes code without declaring a metaphysical break between code and ordinary language. Contracts, laws, forms, and bureaucratic records have always organized action, but executable inscription tightens and scales the circuit between writing and operation.

The machine does not eliminate convention. Executability depends on languages, compilers, runtimes, permissions, hardware, networks, standards, and institutional authorization. A syntactically valid instruction outside its runtime is inert; a script without permission cannot write to the protected directory; a transaction without a recognized account cannot move funds. Computational felicity conditions are built and maintained. The apparent immediacy of execution can therefore hide a broad social and technical apparatus, just as the apparent force of a legal declaration can hide the institution that recognizes the speaker.

Generative AI complicates this relation because the prompt is neither an ordinary performative nor deterministic code. Natural-language instructions are transformed into probabilistic continuations by a model whose behavior depends on training data, architecture, system instructions, retrieval sources, policy layers, conversation history, sampling settings, and interface design. The prompt triggers computation, but its semantic relation to the output is unstable. It can request a genre, role, audience, style, constraint, or procedure without specifying an exact sequence of operations. The system performs a statistically conditioned interpretation of the request.

Calling prompting ‘programming in English’ captures its operational aspiration but obscures the redistributed authorship of the act. Prompt writer, model developer, dataset contributor, platform owner, content moderator, interface designer, and prior user traces participate asymmetrically in the output. The command executes, but not alone; the utterance performs, but without stable felicity conditions. What returns is a provisional outcome embedded in a feedback system. This is why AI literacy cannot stop at prompt technique. The user must learn to read the assembled conditions of action and the consequences they emit.

4. Heidegger: Code as Procedural Revealing

Heidegger’s question concerning technology blocks a purely instrumental interpretation. Technology is not exhausted by devices or by the purposes humans assign them. It is a mode of revealing: a historically specific way in which beings become intelligible and available. Modern technology gathers this revealing as *Gestell*, usually translated as enframing. Enframing orders the real as standing-reserve—resources, energies, capacities, and positions held ready for use. The danger is not simply that machines malfunction or dominate. It is that one mode of revealing may become so comprehensive that other ways of encountering the world disappear from view (Heidegger 1977).

Vee draws on this account to argue that code fosters a procedural understanding of the world. Programming invites phenomena to appear as decomposable operations: inputs, states, variables, branches, functions, outputs, and feedback. The insight is not that everything actually is a procedure. It is that computational practice solicits the world procedurally and builds infrastructures that reward what can be formalized in this manner. A problem becomes a workflow; a relation becomes a graph; a place becomes coordinates; a text becomes tokens; attention becomes engagement; learning becomes measurable activity.

Once code becomes infrastructural, procedural revealing no longer requires that ordinary users write programs. Interfaces translate executable systems into menus, prompts, forms, defaults, recommendations, and conversational exchanges. The procedural work recedes while its categories organize action. Users encounter a friendly surface that has already decided which objects exist, which relations can be expressed, what can be searched, and what counts as completion. In this sense, interface literacy without infrastructural literacy can increase dependence: the surface becomes easier while the conditions of possibility become harder to see.

Generative AI intensifies this concealment by presenting computation in the phenomenological form of another speaker. The system answers, advises, drafts, explains, and summarizes. Conversation lowers the barrier to operation, but it also converts technical mediation into apparent social presence. Beneath the answer lies a massive apparatus for ordering language as data, allocating compute, ranking outputs, filtering requests, and accumulating traces. The conversational interface encourages the user to ask what

the system knows or intends, while the Heideggerian question asks how beings show up when such systems mediate the encounter.

This reframes *Gestell* for educational analysis. Enframing is not another item to place beside teacher, student, chatbot, and rubric. It is the board that positions those pieces and makes their moves intelligible. The assignment reveals the student as an authorial individual; the rubric reveals prose as scorable traits; the platform reveals participation as recorded events; the model reveals language as predictable sequences; the university reveals learning as comparable performance. Metaplay begins when the board becomes sensible as a contingent arrangement rather than the neutral ground of action.

5. Stiegler: Grammatization, Memory, and the Pharmakon

Stiegler radicalizes the relation between humans and technics by refusing to imagine a fully constituted human who later picks up tools. Human becoming and technical exteriorization are co-original. Memory, gesture, language, and knowledge develop through supports outside the organism: tools, marks, images, records, machines, and institutions. These tertiary retentions do not simply store an already complete interior life. They shape what can be remembered, repeated, anticipated, and transmitted (Stiegler 1998).

Grammatization names the process by which temporal and embodied flows are made discrete, spatialized, reproducible, and technically manipulable. Alphabetic writing decomposes speech; notation decomposes music; industrial machinery decomposes gesture; cinema decomposes and recomposes perception; digital systems discretize images, sounds, actions, and relations. Vee mobilizes this lineage to explain code's discreteness and the hybrid ecology in which speech, writing, and computation intermingle without becoming identical. Code is not merely another text. It participates in the ongoing grammatization of action and thought.

The educational importance of grammatization is double. Exteriorization enables memory, critique, inheritance, and collective knowledge. A written argument can be revisited; a procedure can be inspected; a recording can expose what immediate perception missed; a program can be modified and shared. Yet the same discretization can separate knowers from their knowledge. When a process is decomposed and automated, the worker or learner may lose practical understanding even as the system gains repeatability and scale. Technics is therefore pharmacological: poison and cure are not separate properties but competing potentials of the same support.

Generative AI expands grammatization from recorded products toward the statistical formatting of linguistic possibility. Corpora are segmented into tokens; correlations are encoded as parameters; prompts are captured as context; outputs become feedback, evaluation data, and future training material. The system does not store cultural memory like an archive that retrieves stable entries, yet it reorganizes inherited traces into a capacity to generate plausible continuations. Its apparent fluency rests on an industrial memory apparatus whose selections, exclusions, licenses, labor, and energy use are unevenly visible.

Meta-literacy must therefore exceed both skill acquisition and ideological exposure. It requires practical attention to how technical supports compose memory and anticipation. Which histories can the system repeat? Which differences does tokenization preserve or erase? What forms of knowledge become easy to call forth, and which require slow, embodied, local, or relational work? What capacities grow through collaboration with the support, and which atrophy through delegation? These questions make refusal, limitation, and adoption equally available as designed moves rather than moral identities.

6. StudioLab *Perform or Else*: The Performance Paradigm and the Perfumative

Perform or Else identifies performance as a formation that crosses cultural, organizational, and technological domains. Performance can mean a theatrical event, an employee's productivity, or a computer's functionality, yet these uses are not merely homonyms. They converge in a historical demand that subjects and systems continually demonstrate effectiveness under changing criteria. Performance becomes an onto-historical formation: people, organizations, and machines are challenged to perform, to measure the performance, and to transform themselves in response (McKenzie 2001).

The university had entered this paradigm well before generative AI. Students produce assessable outcomes; faculty accumulate publications and evaluations; programs report benchmarks; institutions compete in rankings; platforms record activity; analytics translate learning into dashboards. Performance does not end with an act because the act leaves data that conditions the next act. A submission generates a score; the score enters an aggregate; the aggregate informs a review; the review redirects resources; the redirected resources alter future teaching. The institution performs itself recursively through inscriptions.

Generative AI thickens these circuits. A model's answer is evaluated by the user, logged by the platform, compared against other answers, filtered through safety and quality systems, and folded into product development. A student's AI-assisted draft may be revised, detected, disclosed, graded, discussed, and converted into a precedent for future policy. An instructor's ban may produce concealment, while an invitation may normalize dependence. In each case, the declared performance releases effects that travel beyond the immediate scene.

This article calls that drifting and recursive afterlife *the perfumative*. The term evokes perfume because an act's trace disperses beyond the place of application, mixes with other atmospheres, persists after its source has departed, and affects bodies differently. The performative asks whether the act acquires force; execution asks whether the inscription runs; performance management asks how the result is measured. The perfumative asks what the act releases into the environment and how that residue modifies the conditions of later action.

The perfumative is not a claim that technical effects are ineffable. Traces can often be identified: a search alters recommendations; a prompt changes a conversation; a click updates a profile; an assignment policy changes student tactics; a generated phrase enters another document; a document becomes future training data; a ranking reshapes attention. Yet no inventory captures the whole circuit because effects cross systems and timescales. Perfumative analysis combines empirical tracing with an acknowledgment of incomplete control.

This middle position avoids two symmetrical errors. Technical determinism treats the apparatus as an irresistible cause, while instrumental voluntarism treats it as a neutral tool governed by sovereign intention. The perfumative instead describes distributed action among humans, machines, institutions, interfaces, texts, histories, and environments. Agency remains real but partial. Every move changes the field, and the changed field returns to reorganize the next move.

7. Generative AI as Probabilistic Executable Writing

The sequence from performative to executable to perfumative clarifies what is historically distinctive about generative AI. A language model is not simply a database that returns stored passages, and a prompt is not simply a command that maps deterministically to an output. The system operationalizes natural language inside a probabilistic infrastructure. It converts ordinary expressions of purpose, tone, audience, evidence, style, and constraint into conditions for generating more language. Writing becomes an interface to computation while computation returns as writing.

This arrangement popularizes a peculiar form of coding literacy. Millions of users manipulate system behavior without learning a conventional programming language. They iterate examples, constraints, roles, and evaluation criteria; they observe outputs and adjust inputs; they assemble chains of delegation. Yet conversational ease can conceal the gap between local control and infrastructural control. A user may become highly skilled at steering outputs while remaining unable to inspect the model, dataset, policy stack, commercial incentives, or downstream data practices.

Prompt proficiency is therefore analogous to skilled play. The player learns the affordances, exploits patterns, and develops tactics. Critical AI literacy adds evaluation: it asks about bias, accuracy, privacy, environmental cost, authorship, and appropriate use. Both are necessary, but neither automatically becomes metaplay. The crucial move is positional: the learner must be able to treat the prompt, model, assignment, policy, interface, and evaluation regime as parts of a designed situation whose relations can be changed.

This is also why the language of replacement is inadequate. Asking whether AI replaces the writer assumes that writer and tool are stable units competing for the same task. In practice, the apparatus redistributes tasks and redefines roles. The student becomes verifier, editor, curator, discloser, or clandestine operator; the teacher becomes policy author, detector, activity designer, or collaborator; the platform becomes tutor, ghostwriter, assessor, or archive. The important question is not only who performs the sentence but what configuration produces these identities and consequences.

A probabilistic executable writing environment also multiplies perfumative uncertainty. The same prompt can produce different outputs; a model update can change an established workflow; a policy change can block a legitimate activity; a confident error can travel across documents; a useful scaffold can become a habitual substitution. Meta-literacy is not prediction of every effect. It is disciplined movement between operation, interpretation, tracing, redesign, and replay.

8. Vee's AI-Aware Pedagogy as Metaplay

Vee's AI teaching sequence can be read as a practical curriculum in positional movement. Her January 2025 introduction historicizes the path from GPT-2 to ChatGPT and emphasizes contingency: technologies result from human decisions and need not be treated as inevitable (Vee 2025a). Her account of critical AI literacy then differentiates functional, rhetorical, and ethical capacities, including the ability to use, evaluate, refuse, and assess risks (Vee 2025b). The next step moves from student competence to course architecture by stress-testing assignments and redesigning pathways around learning goals (Vee 2025c).

Stress-testing is deceptively radical. A teacher gives an assignment—or selected component tasks—to an AI system, examines what the system can produce, and asks which operations were carrying the intended learning. The assignment stops functioning as a transparent request and becomes an object of inquiry. Its

theory of authorship, evidence, time, difficulty, originality, and evaluation becomes available for inspection. The teacher briefly occupies the position of student and machine user, then returns with evidence that may require changing the rules.

The 2026 *Norton Guide to AI-Aware Teaching*, coauthored by Vee, Marc Watkins, and Derek Bruff, generalizes this approach. AI awareness begins from course goals, students, and the changing capabilities of the technology; it can support embracing AI, setting boundaries, or finding a balance. The framework resists the binary in which adoption signifies innovation and refusal signifies nostalgia. Both can be pedagogically serious when they are connected to learning and explained as part of a designed environment (Vee, Watkins, and Bruff 2026).

Metaplay extends this pedagogy without replacing it. The player asks, ‘What can the model do?’ The critical player asks, ‘What is the model doing to us and through us?’ The metaplayer asks, ‘What game makes both questions possible, who has been positioned to answer them, and which relations can be altered?’ This third question moves beyond judgment to composition. It may result in a redesigned assignment, a different sequence of human and machine work, a refusal zone, a collective verification protocol, or a changed standard of evidence.

The method remains recursive because a redesign does not solve the apparatus once and for all. An oral defense may privilege confidence or disadvantage some students; a process portfolio may increase surveillance and workload; a mandatory AI exercise may compel participation in a platform a student reasonably rejects; a prohibition may push use underground. AI-aware design becomes metaplay when it includes a performative trace: observe what the intervention releases, invite accounts from differently positioned participants, and revise again.

9. Meta-Literacy as Movement among Positions

Meta-literacy is often defined as reflection on one’s own information practices. Here the term is sharpened through the language of positions. The educational apparatus distributes capacities unevenly: the teacher writes the prompt and rules; the student performs within them; the model produces; the platform records; the rubric classifies; the institution authorizes; the public supplies norms of trust; the archive stores selected traces. None of these roles is absolute, but each organizes what can be seen and done from a location.

A meta-literate learner can move experimentally among these positions without pretending they are interchangeable. From the student position, an assignment can feel like a demand for compliance; from the teacher position, it can appear as a learning pathway; from the model position, it is a pattern of tokens and constraints; from the institution position, it becomes assessable evidence; from a public position, it may carry claims requiring accountability. Movement reveals how a single object changes when the apparatus changes its viewer and stakes.

StudioLab’s D0–D4 heuristic can stage this movement. D0 suspends the demand for a final outside and attends to what exceeds the current frame. D1 treats AI as a tool and asks what operation it performs. D2 centers the responsible subject and asks what the user should do. D3 reads the situated apparatus—its infrastructures, histories, incentives, exclusions, and modes of revealing. D4 experiments with reconfiguration, moving roles, media, constraints, audiences, and evaluation. The sequence is not a ladder of moral superiority; it is a repertoire of positions.

D0 is essential because D4 can easily become a fantasy of mastery. The game designer may believe that making the rules visible is equivalent to controlling every effect. But a redesign enters institutions, encounters bodies, and releases traces that escape intention. Returning to D0 keeps the configuration open to surprise and contestation. The metaplayer has no permanent metaposition, only the capacity to move, listen, trace, and rebuild.

This yields a compact pedagogical cycle: perform → trace → sense → reposition → redesign → replay. Performing creates situated evidence. Tracing follows declared and unintended effects. Sensing includes interpretation, affect, and accounts from different participants. Repositioning changes the viewpoint from which the apparatus is read. Redesign alters relationships, not merely wording. Replay tests the new configuration and begins another cycle. Meta-literacy names the capacity to sustain this cycle; metaplay names its interventionist practice.

10. Heideggerian Engineering: Designing Openings

The phrase Heideggerian Engineering is intentionally tense. Heidegger's analysis cannot be reduced to a design checklist without losing its force, yet contemporary systems cannot wait for ontological criticism to arrive after implementation. The phrase names a practice of building situations that expose and reopen modes of revealing. It asks engineers, teachers, writers, and students to treat disclosure itself as a design concern.

The first question is not 'Does the system work?' but 'What does the system allow to appear?' A writing assistant may reveal prose as a set of improvable patterns while concealing the writer's situated relation to an audience. A rubric may reveal dimensions of performance while concealing interactions among them. An AI tutor may reveal knowledge as available explanation while concealing the practices by which a field establishes and disputes claims. The point is not to reject every reduction; all design selects. The task is to make selections contestable and to preserve pathways toward what they exclude.

A Heideggerian engineering review can ask six practical questions. What entities and roles does the system create? What behaviors does it reward? What becomes measurable or stored? What is rendered standing-reserve? What escapes the frame? How can participants encounter and modify the configuration? These questions translate ontological critique into experimental mechanics without claiming that mechanics exhaust ontology.

For an assignment, the resulting design might require students to compare human and model-generated approaches, document moments of delegation, verify claims with primary sources, interview a reader affected by the text, or redesign the prompt and rubric together. It might also establish non-use spaces where unrecorded conversation, handwriting, observation, or embodied practice can matter. The aim is not maximal complexity. It is a visible relation between means, learning goals, positions, and consequences.

Vee supplies the bridge because literacy describes how technologies become ordinary and transmissible. A technical environment achieves power not only when it is built, but when institutions teach people to inhabit it, reward its fluencies, and forget its contingency. AI-aware pedagogy can therefore become an engineering of openings: a way to prevent computational revealing from appearing as the only imaginable world while still learning how to act competently within it.

11. The Perfumative University

Universities often narrate generative AI as an external disturbance arriving at a previously stable institution. The standard responses—detection, policy, integration, professional development—preserve that story. Yet the university was already a computational performance system. Learning-management systems, citation metrics, rankings, student evaluations, enrollment models, plagiarism services, assessment dashboards, productivity measures, and algorithmic admissions had long organized academic life. AI does not invade a pretechnical university; it reveals and accelerates the university's existing technical constitution.

The institution can be read as an executable writing environment. Forms call databases; rubrics produce scores; scores produce classifications; classifications produce resources; resources reorganize programs and subjects. Course descriptions, policies, records, and rankings do not merely represent the university. They help perform it. When these inscriptions become automated and generative, they also produce perfumative traces that circulate across offices, platforms, and futures. A small local optimization can reorganize who is noticed, supported, suspected, or excluded.

This perspective changes what an AI strategy should contain. Training people to use approved tools and drafting integrity rules are insufficient because they treat the apparatus as a collection of user choices. A meta-literate institution would also map data flows, vendor dependencies, labor shifts, environmental costs, accessibility effects, changes in disciplinary judgment, and the feedback loops created by new assessments. It would make room for refusal and experimentation while demanding that each intervention be traced after deployment.

The perfumative university is not merely a dystopian label. Perfumative traces can also support collective memory, solidarity, access, and invention. A redesigned assignment can circulate as a shared resource; a transparent disclosure practice can create trust; a student critique can alter procurement; a refusal can preserve a form of attention; a model comparison can expose hidden standards. The same technical supports remain pharmacological. What matters is whether participants can sense and reconfigure their effects.

Vee's contribution is a pedagogical bridge between literacy studies and this institutional critique. Literacy joins sign, infrastructure, practice, value, and transmission. It shows how computational environments become ordinary not only through engineering but through teaching. To move from AI literacy to meta-literacy is therefore to teach participants to read systems of inscription, execution, performance, feedback, and disclosure—and to intervene without imagining that intervention ends the story.

Conclusion: Play the Revealing

Reading Annette Vee as a metaplayer clarifies the continuity between her history of coding literacy and her AI-aware pedagogy. The first demonstrates that code becomes socially consequential as it moves from specialist practice to infrastructural presupposition. The second shows teachers how to historicize AI, develop critical capacities, stress-test assignments, and redesign pathways around learning goals. Together they describe a movement from use to awareness to transformation.

Austin explains how language acts through assembled conditions; Vee identifies code's executable double address; Heidegger shows that execution is already a mode of revealing; Stiegler shows how technical inscription composes memory, knowledge, and loss; McKenzie shows performance becoming a cross-domain demand; the perfumative names the recursive traces those performances release. Generative AI condenses this genealogy by making natural language both the medium of instruction and the generated product of computation.

The resulting pedagogy does not ask students merely to prompt better or to denounce the system from a safe distance. It cultivates movement among player, critic, designer, and witness. It asks what the board reveals, who is positioned where, what becomes standing-reserve, what the move releases, and how the configuration can be replayed. Meta-literacy is the capacity for that movement; metaplay is its practice.

No metaplayer controls the entire game. Every redesign generates another atmosphere of expectations, exclusions, affordances, and traces. The educational task is therefore recursive rather than terminal: perform, trace, sense, reposition, redesign, replay. The strongest command the suite can offer is also the simplest: play the revealing.

Terms Introduced or Extended Here

Metaplayer and perfumative are interpretive terms developed by this article. They are not presented as Vee's own vocabulary. Likewise, the D0–D4 sequence and Heideggerian Engineering are StudioLab heuristics used to translate the theoretical argument into pedagogical practice.

References

- Austin, J. L. 1962. *How to Do Things with Words*. Cambridge, MA: Harvard University Press.
- Banks, Adam J. 2006. *Race, Rhetoric, and Technology: Searching for Higher Ground*. Mahwah, NJ: Lawrence Erlbaum.
- Derrida, Jacques. 1976. *Of Grammatology*. Translated by Gayatri Chakravorty Spivak. Baltimore: Johns Hopkins University Press.
- Frabetti, Federica. 2021. "Review of Annette Vee, Coding Literacy." *Computational Culture* 8. [source](#)
- Heidegger, Martin. 1977. "The Question Concerning Technology." In *The Question Concerning Technology and Other Essays*, translated by William Lovitt, 3–35. New York: Harper & Row.
- McKenzie, Jon. 2001. *Perform or Else: From Discipline to Performance*. New York: Routledge. [source](#)
- Stiegler, Bernard. 1998. *Technics and Time, 1: The Fault of Epimetheus*. Translated by Richard Beardsworth and George Collins. Stanford, CA: Stanford University Press. [source](#)
- Vee, Annette. 2013. "Understanding Computer Programming as a Literacy." *Literacy in Composition Studies* 1 (2): 42–64. [source](#)
- Vee, Annette. 2017. *Coding Literacy: How Computer Programming Is Changing Writing*. Cambridge, MA: MIT Press. [source](#)
- Vee, Annette. 2025a. "Introducing AI & How We Teach Writing." January 30. [source](#)
- Vee, Annette. 2025b. "What Is Critical AI Literacy?" February 20. [source](#)
- Vee, Annette. 2025c. "AI-Aware Teaching Examples." March 24. [source](#)
- Vee, Annette, Marc Watkins, and Derek Bruff. 2026. *The Norton Guide to AI-Aware Teaching*. New York: W. W. Norton. [source](#)

Play the Revealing

Annette Vee after AI Literacy

Vee → coding literacy → executable writing → revealing → grammatization → perfumative → metaplay

CENTRAL MOVE AI literacy teaches people to act within an AI-saturated environment. Meta-literacy teaches them to perceive, test, and redesign the environment itself.

The Wrong First Question

How do we teach students to use AI responsibly? The question now organizes faculty meetings, syllabus revisions, assessment workshops, and anxious conversations about the future of writing. It is not a bad question. It is simply not the first one. Before anyone decides whether a student may use a model to brainstorm, summarize, draft, or revise, a whole educational game has already been built. The teacher assigns; the student produces; the rubric classifies; the platform records; the institution certifies; the model waits in another browser tab. Responsibility enters only after these positions have been made to seem natural.

Annette Vee gives us a way to rewind the scene. Her importance is not that she supplies the definitive rule for ChatGPT. She has consistently resisted that kind of answer. Her work asks how a technical practice becomes ordinary enough to count as literacy, how institutions begin presupposing it, and how teachers can redesign their environments when the practice changes. Read across *Coding Literacy* and her AI-aware teaching work, Vee becomes less a literacy advocate than a metaplayer: a scholar who moves between the rules, infrastructures, positions, and redesigns that make literacy possible.

A player asks what AI can do. A critical player asks what AI is doing to us. A metaplayer asks what game makes both questions possible—and how the game might be altered. That final question matters because critique alone can become a spectator sport. One can denounce bias, labor extraction, hallucination, surveillance, and ecological cost while leaving the assignment, platform, evaluation scheme, and distribution of authority exactly as they were. Metaplay turns diagnosis into situated recomposition.

Vee's Long Game: Literacy Becomes an Environment

Vee defines literacy socially rather than privately. A literacy is not just a competence possessed by an individual; it is a valued set of practices tied to infrastructural communication technologies. Institutions organize around it. They distribute access through it. Eventually they presuppose it. Textual literacy took shape through schools, churches, governments, markets, archives, and households. Coding follows a related but not identical trajectory as software travels from specialized domains into the ordinary environments of work, communication, mobility, finance, and education (Vee 2013; 2017).

This history produces an awkward present: code is infrastructural while the ability to read and write code is not widely distributed. Most people do not program, but nearly everyone is programmed around. Interfaces turn prior executable writing into menus, feeds, profiles, forms, permissions, recommendations, and prompts. We write through systems whose writing we cannot ordinarily inspect. The surface says ‘compose’; the infrastructure decides what can be composed, stored, found, ranked, and monetized.

Calling coding a literacy is therefore political. The term carries a moral charge: literacy promises participation and competence, while illiteracy marks deficiency. Coding-for-all campaigns can widen access, but they can also shift a structural demand onto individuals—learn to code or accept your obsolescence. Vee keeps both possibilities open. Her strongest concept is transformative access: not simply entry into a system, but the practical ability to understand its operations, expose its values, and alter its interfaces (Frabetti 2021).

Transformative access is already a theory of metaplay. The user learns the controls. The literate user understands the conventions. The critic reads the ideology. The metaplayer can change position and reorganize the relation among controls, conventions, institutions, and consequences. This does not mean gaining root access to the whole world. It means refusing to confuse the visible interface with the field of action.

When Writing Executes

Austin taught us to notice that language does things. A promise, verdict, declaration, or bet does not merely report an event; under the right conditions, it performs one. Vee’s work on code makes that operational dimension technical. Code addresses human readers as meaningful inscription and machines as executable instruction. A program can describe a procedure and initiate it. Writing sorts the list, opens the gate, transfers the money, denies the request, renders the image, or calls another process.

Execution is never magic. It depends on a runtime, permissions, standards, hardware, networks, and institutions. Code has felicity conditions too. The line runs because a thick apparatus recognizes it. But the tighter circuit between inscription and operation makes an old property of writing visible: documents have always acted through arrangements of authority and infrastructure. Executable writing scales and accelerates that action.

Generative AI makes this arrangement stranger. A prompt is not deterministic code, yet it triggers computation. It may specify a role, style, audience, genre, example, prohibition, or goal while leaving the sequence of operations unresolved. The model responds probabilistically according to parameters, context, training, retrieval, policies, and interface. Natural language becomes an operational surface, and computation returns as natural language. The user feels less like a programmer than a conversational partner, even as the conversation activates an industrial apparatus.

Prompting is thus a new popular encounter with executable writing, but prompt literacy is not meta-literacy. Better prompts make better players. They do not necessarily reveal who built the board, what the board measures, or how a successful move changes the next round. The rhetoric of ‘AI fluency’ can turn a historically contingent platform arrangement into a personal employability requirement. Vee’s literacy history teaches us to be suspicious whenever a technical demand arrives wearing literacy’s halo.

Play the Revealing

Heidegger's question concerning technology shifts attention from devices to disclosure. Technology is a way the world shows up. Modern technology gathers things as standing-reserve: resources, capacities, energies, and positions held ready for use. Vee applies this insight to code's procedurality. Under computational revealing, problems appear as workflows, relations as graphs, places as coordinates, writing as tokens, attention as engagement, and learning as measurable performance.

That does not mean the categories are false. Procedures, graphs, coordinates, tokens, and measures can be extraordinarily useful. The danger comes when usefulness presents itself as the only way anything can appear. A student becomes a stream of submissions and scores. A teacher becomes content delivery and feedback capacity. A text becomes an input available for transformation. An archive becomes training material. A language becomes token probabilities. The world becomes ready-to-process.

This is why *Gestell* should not be another box in an AI-literacy diagram. It is the board that positions all the boxes. Teacher, student, model, platform, rubric, and institution become pieces whose identities make sense inside an ordering. The metaplayer does not leap off the board into purity. The metaplayer makes the board perceptible, moves among positions, and tries arrangements that disclose something else.

Stiegler helps explain the stakes of those arrangements. Technical supports exteriorize memory, gesture, and knowledge. Grammatization makes flows discrete and reproducible: speech becomes letters, music becomes notation, gesture becomes industrial sequence, perception becomes recorded image, language becomes data. Such exteriorization is pharmacological. It can support thought, inheritance, and collective memory, or dispossess people of capacities by turning knowledge into automated procedure. AI is neither cure nor poison in advance. Its educational effects emerge from the configuration.

From Performance to Perfumative Trace

Perform or Else describes a culture in which people, organizations, and machines must continually demonstrate effectiveness. Universities inhabit this performance paradigm through outcomes, rubrics, rankings, evaluations, dashboards, productivity measures, and platform analytics. Generative AI did not bring performance management to education. It arrived inside an institution already trained to make learning visible as comparable results.

Performance now curls back on itself. An act creates data; the data changes the system; the changed system reorganizes the next act. A student submits a paper, receives a score, alters strategy, and contributes to an aggregate that may reshape the course. A model returns an answer, the user evaluates it, the platform logs the interaction, and future versions change. The declared act ends, but its residue keeps moving.

The perfumative names that moving residue. Like perfume, a technical act disperses beyond its point of application, mixes with other atmospheres, persists after the source leaves, and affects bodies unevenly. A search changes recommendations. A prompt changes a conversation. A policy changes tactics. A generated phrase migrates into another document. A redesign changes what students believe teachers value. Performance asks whether an act works; the perfumative asks what the act releases.

This distinction keeps metaplay honest. Redesign is not redemption. An oral exam may privilege quick confidence. A process portfolio may become surveillance. A mandatory AI exercise may compel students to enter a platform they reject. A ban may convert honest inquiry into concealment. Every intervention

needs a trace phase: who experienced what, what became newly visible, what was lost, and what should be replayed differently?

The Teacher as Epistemic Game Designer

Vee's AI-aware teaching practice already performs this turn. Historicize the technology so it does not appear inevitable. Develop functional, rhetorical, and ethical capacities. Begin from learning goals. Stress-test the assignment with AI. Distinguish the operations students need to practice from those that can be delegated or contested. Choose among resistance, limited adoption, and integration according to the educational situation (Vee 2025a; 2025b; 2025c; Vee, Watkins, and Bruff 2026).

Stress-testing changes the status of the assignment. The teacher temporarily stops treating it as a transparent request and plays it from the position of an AI-mediated student. What does the task reward? Which shortcuts are invitations built into the prompt? Where is the actual learning? What theory of authorship is hidden in the submission format? Returning to the teacher position with that evidence, the instructor can alter roles, media, audiences, constraints, sources, sequence, and evaluation.

The teacher becomes a designer of epistemic games. Expertise does not disappear; it becomes more demanding. The designer must understand the disciplinary game well enough to change it without pretending that change guarantees emancipation. Students need room to become designers too: to explain why a tool belongs in one stage and not another, to challenge the rubric, to compare systems, to document uncertainty, and to propose forms of evidence the assignment had excluded.

The resulting protocol is simple enough to remember: perform → trace → sense → reposition → redesign → replay. Perform the task from more than one position. Trace what the performances produce and leave behind. Sense the experience through different participants. Reposition the player, critic, teacher, system, and audience. Redesign the relations rather than merely decorating the prompt. Replay and read the new trace.

The player asks: *What can AI do?* The critic asks: *What is AI doing to us?* The metaplayer asks: *What game makes both questions possible, and how can we alter it?* That is the move from literacy to meta-literacy. It does not end the game. It gives us a way to keep the revealing in play.

Short Works Cited

- Austin, J. L. 1962. *How to Do Things with Words*. Cambridge, MA: Harvard University Press.
- Frabetti, Federica. 2021. "Review of Annette Vee, *Coding Literacy*." *Computational Culture* 8. [source](#)
- Heidegger, Martin. 1977. "The Question Concerning Technology." In *The Question Concerning Technology and Other Essays*, translated by William Lovitt, 3–35. New York: Harper & Row.
- McKenzie, Jon. 2001. *Perform or Else: From Discipline to Performance*. New York: Routledge. [source](#)
- Stiegler, Bernard. 1998. *Technics and Time, 1: The Fault of Epimetheus*. Translated by Richard Beardsworth and George Collins. Stanford, CA: Stanford University Press. [source](#)
- Vee, Annette. 2017. *Coding Literacy: How Computer Programming Is Changing Writing*. Cambridge, MA: MIT Press. [source](#)
- Vee, Annette. 2025a. "Introducing AI & How We Teach Writing." January 30. [source](#)
- Vee, Annette. 2025b. "What Is Critical AI Literacy?" February 20. [source](#)
- Vee, Annette. 2025c. "AI-Aware Teaching Examples." March 24. [source](#)
- Vee, Annette, Marc Watkins, and Derek Bruff. 2026. *The Norton Guide to AI-Aware Teaching*. New York: W. W. Norton. [source](#)

BAD READERS NO. V

DON'T LEARN AI. PLAY IT.

Annette Vee's Meta-Literacy Machine

AI literacy has been kidnapped by the checklist: write a better prompt, disclose your use, verify the output, protect your data, become more productive. These are useful tactics, but tactics are not freedom. They teach you to survive inside a system whose rules, roles, and rewards remain untouched.

X ~~PROMPT LIKE A PRO~~

X ~~CATCH THE CHEATER~~

X ~~AUTOMATE THE HOMEWORK~~

X ~~CALL IT RESPONSIBLE~~

X ~~NEVER QUESTION THE BOARD~~

Vee's wager is sharper: literacy becomes political when institutions presuppose a technical practice. Code now underwrites writing; AI now rewrites the conditions of both. Do not mistake competent use for comprehension of the game.

EXIT SIGN

The player asks what AI can do. The critic asks what AI is doing to us. **The metaplayer asks what game makes both questions possible—and changes the game.**

LITERACY IS NOT A CHECKLIST.

THE RULES ARE PART OF THE READING.

SIDE A / INSCRIPTION MUTATES

EXECUTE!**Austin -> code -> Vee -> probabilistic command**

Austin broke the idea that language only describes. A promise promises; a verdict sentences; a declaration institutes. Words take force because a scene of conventions, authorities, participants, and procedures recognizes the act.

CODE'S DOUBLE LIFE

Code is readable description for people and executable instruction for machines. It can sort, render, open, deny, transfer, score, and call another process. The line acts only because a runtime, permission structure, language, and institution let it run.

TYPE --> CODE --> PROMPT --> OUTPUT --> TRACE

A prompt is not deterministic code. It triggers a probabilistic system whose response depends on model weights, training data, system instructions, policies, retrieved material, conversation history, and interface. Natural language becomes a handle on computation, while computation returns wearing the face of natural language.

THE GLITCH IN 'AI FLUENCY'

Better prompting makes better players. It does not automatically reveal the infrastructure, the ownership, the hidden instructions, the labor, the energy, the evaluation regime, or the future uses of the trace. **Execution is never just you and the machine.**

Vee calls coding a literacy because it has become socially useful, institutionally valued, and infrastructural. That means literacy is not a badge you wear inside your skull. It is a field that assigns access, competence, legitimacy, and blame.

WRITING THAT RUNS**RUNS THROUGH INSTITUTIONS, INTERFACES, AND YOU.**

SIDE B / THE BOARD APPEARS

WHO'S PLAYING WHOM?

Gestell is the board—not another box

Heidegger's question is not whether the tool is good or bad. Technology is a mode of revealing: a way the world becomes available. Modern technology orders beings as standing-reserve—resources, scores, capacities, data, energy, and attention waiting to be used.

STUDENT	TEACHER	MODEL	UNIVERSITY
submission / profile / risk	policy / prompt / evaluator	generator / tutor / suspect	platform / metric / certificate
RUBRIC	LMS	VENDOR	PUBLIC
traits / score / norm	record / dashboard / gate	interface / log / contract	trust / archive / consequence

GESTELL / THE GRID THAT MAKES THESE POSITIONS LOOK NATURAL

PERFUMATIVE SNIFF TEST

The act ends. The trace keeps moving. A prompt changes the conversation; the conversation changes the draft; the draft changes the grade; the grade changes the tactic; the tactic changes the next assignment. Performance asks whether it worked. **The perfumative asks what it released.**

No one player owns the whole circuit. Humans act, systems act, policies act, metrics act, archives act. Metaplay starts when the board becomes perceptible enough to bend.

THE BOARD PLAYS BACK.

EVERY MOVE CHANGES THE NEXT MOVE.

FINAL SIDE / DO THE MOVE

METAPLAY!

Play -> trace -> sense -> reposition -> redesign -> replay

Metaplay is not escape to a pure outside. It is disciplined movement among positions, followed by a change in relations. Use the system, criticize it, refuse it, reroute it, redesign it—and then read what your intervention did.

D0	SUSPEND — notice what exceeds the current frame
D1	OPERATE — learn what the tool actually does
D2	ANSWER — take responsibility for the situated choice
D3	REVEAL — map infrastructure, value, labor, and position
D4	REMAKE — alter roles, rules, media, audience, evaluation

Try it on one assignment tonight. Run the task normally. Ask an AI to attempt it. Identify what the assignment assumes about authorship, evidence, labor, and originality. Ask what becomes visible and what becomes standing-reserve. Change one relation—not just one sentence—and replay.

LAST COMMAND

You do not need a better user manual. You need the capacity to move among user, critic, designer, refuser, witness, and learner. **PLAY THE REVEALING.**

Source trail: Annette Vee, *Coding Literacy* (2017); Vee, “What Is Critical AI Literacy?” and “AI-Aware Teaching Examples” (2025); Austin, Heidegger, Stiegler, and McKenzie. Metaplay and perfumative are the zine’s StudioLab extensions.

PLAY THE REVEALING.

THEN TRACE WHAT YOUR PLAY RELEASED.

STUDIOLAB TEACHING / GAME KIT

THE VEE MACHINE aka

ShuttleKong: Ride the Drives, Mind the V

A six-round exercise for turning one assignment into a playable apparatus

CORE PROPOSITION AI-aware teaching becomes meta-literacy when participants can move among positions, inspect the game, redesign its relations, and trace what the redesign releases.

What this kit does

Participants bring one real assignment and pass it through six positions: player, AI player, critic, Heideggerian engineer, metaplayer, and perfumative tracer. The point is not to force adoption or refusal. The point is to make the educational configuration visible enough to choose and redesign deliberately.

ROUND	POSITION	PRIMARY MOVE
1	Player	Perform the assignment as currently written.
2	AI Player	Test multiple model attempts and preserve the traces.
3	Critic	Expose assumptions about authorship, evidence, labor, and value.
4	Heideggerian Engineer	Read what the apparatus reveals and renders standing-reserve.
5	Metaplayer	Change roles, media, constraints, audiences, and evaluation.

Round 6, the Perfumative Trace, is the replay mechanism: map intended and unintended effects, then decide what the next version must change.

Format

90 minutes in one session, or six shorter studio rounds. Works with faculty, students, designers, writing-program teams, or mixed groups. Use the pages as a facilitator script and writable worksheet.

BEFORE PLAY

SET THE TABLE

Choose the object, disclose the constraints, and protect participant agency

Materials

- One existing assignment prompt, rubric, and any model/example submission.
- Access to one or more generative AI systems, if participants elect to use them.
- A shared trace space: paper, whiteboard, or a document visible to the group.
- This kit, writing tools, and a timer.

Participation and data boundaries

Do not require participants to enter personal, protected, unpublished, or student-identifiable material into a commercial model. Offer a no-account pathway: the facilitator can provide pre-generated outputs or use a fictionalized assignment. Refusal is a legitimate move, not a failure to participate.

Assignment / activity being tested

Stated learning goal: what should participants become able to do or understand?

Current deliverable, audience, time frame, and evaluation method

Position tokens

Assign or rotate six voices: student, teacher, AI system, institution, public/audience, and archive/future use. A position is not a caricature; it is a prompt to notice how the same assignment appears under different stakes.

FACILITATOR RULE Separate description from judgment. First record what the apparatus does; then argue about whether the effect is desirable.

ROUNDS 1-2

PLAY IT / LET THE MODEL PLAY IT

Generate situated evidence before interpretation

ROUND 1: PLAYER

10 MIN

Inside the current rules

PURPOSE Complete or simulate the task as written. Notice where effort, judgment, uncertainty, and disciplinary knowledge actually enter.

1. Underline every operation the assignment requires.
2. Circle moments where the instructions leave room for judgment.
3. Mark what you believe the rubric will reward, regardless of the stated goal.

Operations performed by the human player

Where did learning feel most likely to occur?

ROUND 2: AI PLAYER

15 MIN

Across the interface

PURPOSE Ask at least two model attempts to perform the whole task or selected operations. Change one condition between attempts and retain the outputs, prompts, date, and system name.

4. Use the assignment exactly as written for attempt A.
5. For attempt B, add the rubric or a model and note what changes.
6. Record refusals, uncertainty, hallucinations, bland competence, and surprising strengths.

What could the AI perform easily, unevenly, or not at all?

What hidden instruction did the AI seem to infer?

DO NOT SCORE YET The output is evidence about the assignment–model configuration, not proof that ‘AI can’ or ‘AI cannot’ perform the discipline.

ROUND 3 / CRITIC

READ THE GAME

Expose the assignment's theory of intelligence, authorship, and value

ROUND 3: CRITIC

15 MIN

Across several positions

PURPOSE Interpret what the player and AI-player traces reveal about the current game. The target is the configuration, not the moral worth of an individual user.

7. Name the assignment's implicit author: solitary creator, collaborator, curator, verifier, performer, or something else.
8. Identify what counts as evidence and who has authority to validate it.
9. Locate invisible labor: source creation, feedback, care, moderation, data work, and technical maintenance.

DIMENSION	CURRENT ASSUMPTION	WHAT THE AI TEST EXPOSED
Authorship	_____	_____
Evidence	_____	_____
Originality	_____	_____
Labor / time	_____	_____
Expertise	_____	_____

Who gains room to act under the current rules? Who carries added risk or work?

What should remain a human practice even if a model can imitate the deliverable? Why?

CRITICAL THRESHOLD Do not confuse a convincing product with the learning process the product was meant to evidence.

ROUND 4 / HEIDEGGERIAN ENGINEER

PLAY THE REVEALING

Treat Gestell as the board that positions the pieces

ROUND 4: HEIDEGGERIAN ENGINEER

15 MIN

At the level of disclosure

PURPOSE Ask what becomes visible, calculable, and available through the assignment—AI configuration. Then ask what the frame makes difficult to encounter.

10. List the entities the system creates: user, author, source, score, attempt, risk, evidence.
11. Name what becomes standing-reserve: attention, language, student work, feedback, compute, or time.
12. Identify one excluded mode of knowing: embodied, tacit, local, relational, historical, slow, or uncertain.

The assignment reveals learning as...

The AI interface reveals writing / knowledge / the student as...

What is made easy to measure or store? What is made hard to perceive?

One opening: a practice or relation that could disclose the object otherwise

BOARD TEST Gestell is not an evil device beside the other pieces. It is the ordering that lets the pieces appear as teacher, student, model, resource, score, and output.

ROUND 5 / METAPLAYER

CHANGE ONE RELATION

Redesign the game, not merely the wording of the prompt

**ROUND 5:
METAPLAYER**

20 MIN

Between game and metagame

PURPOSE Produce a playable redesign that changes at least three design dimensions while preserving or deliberately revising the learning goal.

- 13. Change a role: who asks, judges, verifies, or addresses whom?
- 14. Change a medium or sequence: conversation, annotation, field observation, oral defense, public artifact, code, image, or staged replay.
- 15. Change a constraint, audience, evidence rule, or evaluation method.

DESIGN DIMENSION	CURRENT GAME	REDESIGNED GAME
Roles		
Sequence		
Media		
Evidence		
Audience		
Evaluation		

New assignment move in 60 words or fewer

Why this redesign better aligns means, learning, and accountability

REFUSAL IS DESIGN A no-AI zone can be a strong metaplay move when it creates a protected form of attention or practice and explains the educational reason.

ROUND 6 / PERFUMATIVE TRACE

WHAT DID THE MOVE RELEASE?

Map effects that exceed the declared purpose

ROUND 6: PERFUMATIVE TRACE

15 MIN

After and around the act

PURPOSE Forecast and then observe how the redesign changes tactics, attention, labor, trust, data, participation, and future expectations.

16. Separate intended outcomes from plausible side effects.
17. Trace who experiences each effect and on what timescale.
18. Choose what evidence will tell you whether to keep, revise, or reverse the redesign.

TRACE	POSSIBLE EFFECT	WHO / WHERE	EVIDENCE TO COLLECT
Learning	_____	_____	_____
Tactic	_____	_____	_____
Labor	_____	_____	_____
Trust / risk	_____	_____	_____
Data / future	_____	_____	_____

Most promising unintended effect

Most concerning unintended effect

Reversal condition: what evidence would make us stop or undo the redesign?

PERFUMATIVE RULE A successful redesign is not one with no side effects. It is one whose effects can be sensed, discussed, and revised without hiding their distribution.

DEBRIEF / REPLAY

TURN THE ROUND INTO A PRACTICE

Consolidate the learning, assign the next test, and keep the metagame open

Debrief in four voices

PLAYER	What became easier or harder to do? _____
CRITIC	Which rule or assumption became newly visible? _____
METAPLAYER	Which relationship changed most consequentially? _____
WITNESS	What trace should the group keep watching? _____

Version 1 to test: exact change, owner, setting, and date

Evidence review: when and with whom will the traces be read?

D0–D4 positional check

D0 suspend mastery; D1 operate the tool; D2 take situated responsibility; D3 reveal infrastructure and position; D4 reconfigure the game. Do not remain at D4 as if redesign were a sovereign outside. Return to D0, attend to what escaped the frame, and replay.

EXIT LINE The player asks what AI can do. The critic asks what AI is doing to us. The metaplayer asks what game makes both questions possible—and how to alter it.

Source trail

The exercise adapts Annette Vee’s account of coding literacy and her AI-aware practice of historicizing AI, aligning use or refusal with learning goals, and stress-testing assignments (Vee 2017; 2025a–c; Vee, Watkins, and Bruff 2026). The terms *metaplay*, *perfumative trace*, D0–D4, and *Heideggerian Engineering* are StudioLab extensions developed for this suite.

STUDIOLAB / PORTABLE DOCTRINE

TWELVE THESES ON AI META-LITERACY

Vee -> executable writing -> revealing -> grammatization -> perfumative -> metaplay

These theses extend Annette Vee's movement from coding literacy to AI-aware teaching. *Metaplayer* and *perfumative* are StudioLab terms: the first names positional redesign, the second the traces that escape and recursively alter a technical performance.

1

AI literacy is already insufficient.

Functional competence matters, but it arrives after systems have assigned the user a role and defined success. Meta-literacy begins when those conditions become readable and alterable.

2

Prompting is skilled play, not metaplay.

A better prompt can steer an output without exposing the model, platform, data, policy, or institution that makes the exchange possible. Fluency inside the interface is not knowledge of the game.

3

Every interface assigns positions.

Teacher, student, author, source, model, evaluator, and public are not neutral identities waiting outside the apparatus. They are produced and constrained by permissions, workflows, genres, and stakes.

4

Every assignment contains a theory of intelligence.

Its tasks and rubric decide what counts as knowing, creating, remembering, arguing, and learning. AI makes that theory visible by imitating the deliverable and separating product from practice.

5

Executable writing turns literacy into infrastructure.

Code does not only represent action; under assembled conditions it initiates operations. Generative AI extends this operability by making natural language a probabilistic interface to computation.

6

Gestell is the board, not another piece.

Enframing is the ordering that lets language, attention, students, sources, and compute appear as available resources. Metaplay starts when the ground of the game becomes perceptible as a contingent design.

THESES 7–12

FROM CRITIQUE TO REPLAY

There is no sovereign outside—only a more capable movement through the game

7

Grammatization is pharmacological.

Technical inscription can preserve memory, enable critique, and widen participation; it can also decompose knowledge into automatable units and dispossess people of capacities. Adoption and refusal must be designed, not worn as identities.

8

Critique without redesign remains inside the game.

Naming bias, extraction, hallucination, and surveillance is necessary but incomplete when assignments, platforms, and evaluation schemes stay intact. Meta-literacy composes alternatives and tests them.

9

Refusal counts as a move.

A no-AI practice can protect attention, embodiment, relationship, privacy, or disciplinary formation. Refusal becomes metaplay when it explains the learning it protects and remains open to evidence about its effects.

10

Every intervention leaves a perfumative trace.

A performance ends; its residue circulates through data, tactics, expectations, policies, and future designs. The metaplayer asks not only whether the move worked, but what it released and for whom.

11

Meta-literacy teaches movement among positions.

The learner must practice being user, critic, designer, refuser, verifier, audience, and witness. No single position supplies the whole apparatus, and no metaposition grants sovereign control.

12

Play the revealing.

Perform, trace, sense, reposition, redesign, and replay. Education should not merely train people for an AI-saturated game; it should cultivate the collective capacity to make and remake the conditions of play.

THE METAPLAYER'S LOOP PLAY -> TRACE -> SENSE -> REPOSITION -> REDESIGN -> REPLAY

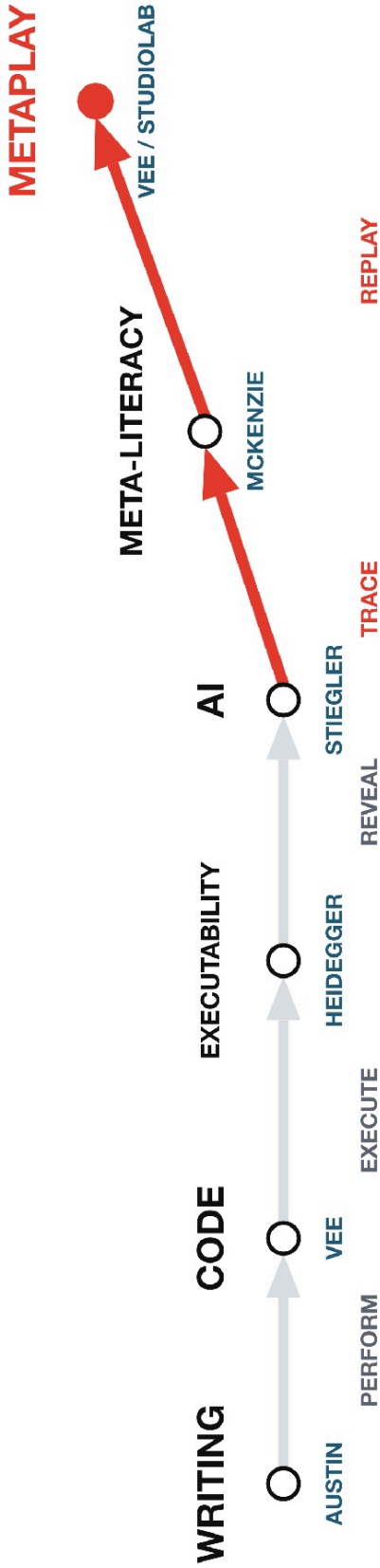
Source trail: Annette Vee, *Coding Literacy* (2017); “What Is Critical AI Literacy?” and “AI-Aware Teaching Examples” (2025); Vee, Watkins, and Bruff, *The Norton Guide to AI-Aware Teaching* (2026); Austin; Heidegger; Stiegler; McKenzie.

VI. From Literacy to Metaplay

ANNETTE VEE AS METAPLAYER / A STUDIOLAB COSMOGRAM

FROM LITERACY TO METAPLAY

Writing becomes executable; AI regrammatizes it; the apparatus becomes playable.



GESTELL / THE BOARD THAT POSITIONS THE PLAYERS

The board orders student, teacher, model, language, attention, and evidence as available positions.

Metaplay bends the board—then reads the trace left by the bend.

PLAY → TRACE → SENSE → REPOSITION → REDESIGN → REPLAY

THE PLAYER USES THE SYSTEM. THE CRITIC READS IT. THE METAPLAYER RECONFIGURES IT.

Writing becomes executable; AI regrammatizes it; the apparatus becomes playable.

APPENDIX A. VEE WITH/FOR NORTON

Substack as AI-Aware Teaching Infrastructure

Annette Vee's AI & How We Teach Writing is not merely a side channel for promoting a finished book. It is a publishing-and-teaching apparatus in its own right: a recurrent, public, revisable space in which research, classroom experiments, institutional questions, publisher reach, and teacher response can meet. The newsletter announces itself as a collaboration between W. W. Norton and Vee, but its address is consistently to teachers and their students. It therefore gives the phrase with and for a useful double force. Vee writes with a publisher, yet she writes for a distributed pedagogical public whose problems, refusals, and experiments continually exceed the book as a bounded object.

From Book to Feed—and Feed to Book

The usual publication sequence runs from research to manuscript to book, followed by promotion. The Substack scrambles that order. Timely posts explain a concept, invite classroom use, collect examples, and make the evolving terrain discussable before and after the guide appears. The feed is not simply preliminary material and the book is not simply the feed made permanent. They operate as different temporalities of the same project: the newsletter moves at the speed of classroom change, while The Norton Guide to AI-Aware Teaching consolidates durable pedagogical principles, frameworks, and examples.

This is especially visible in the movement from the newsletter's opening account of AI and writing to posts on critical AI literacy, assignment redesign, policy, student use, and academic integrity. Each post isolates a pressure point that teachers can recognize immediately. The guide then reorganizes those pressures into a course-scale sequence: learning goals, students, capabilities, roles for AI, assessments, policies, activities, reading, writing, research, tutoring, integrity, and collaboration. The feed works episodically; the guide works architecturally. Together they make AI-aware teaching both timely and teachable.

With Norton, for Teachers

The partnership matters because publishers are not neutral pipes. Norton supplies editorial labor, distribution, professional-development networks, adoption pathways, and a recognizable address to instructors across disciplines. Those resources widen the field of encounter beyond writing studies. At the same time, they situate the project inside the commercial and institutional machinery of educational publishing: textbook adoptions, instructor access, sales representatives, campus reading groups, events, and ancillary resources. To say that Vee is substacking with Norton is therefore to name both collaboration and infrastructure.

To say that she is substacking for AI-aware teachers names the counter-direction of that infrastructure. The posts do not merely tell instructors which tool to adopt. They begin from teaching conditions: limited time, uneven authority, different disciplines, different students, and legitimate positions ranging from resistance to engagement. The publisher amplifies the address, but the pedagogical unit remains the situated course. This keeps the project from collapsing AI awareness into product awareness. Its central question is not which platform should win; it is what students are meant to learn and what arrangement of people, tasks, tools, evidence, and boundaries can support that learning.

AI-Aware Teaching as Positional Play

Vee's account of critical AI literacy distinguishes functional, rhetorical, and ethical capacities. The sequence matters. Functional competence asks what a system can do and how to operate it. Rhetorical judgment asks how outputs fit audiences, purposes, values, and situations. Ethical judgment asks what risks, exclusions, costs, and responsibilities travel with the operation. AI-aware teaching adds an institutional turn: those judgments must be composed into assignments, policies, conversations, assessments, and course designs.

The Norton guide's refusal of a single pro- or anti-AI stance is not indecision. It is a positional method. An instructor may resist AI in one task, allow it under conditions in another, and use it as an object of analysis elsewhere. What matters is whether the position follows from learning goals, student needs, and an explicit account of consequences. This is already close to metaplay because the teacher is not restricted to making better moves inside one fixed policy. The teacher can change the role of AI, the sequence of work, the evidence of learning, the terms of disclosure, and the relation between classroom and platform.

Yet AI awareness and metaplay are not identical. AI-aware teaching can remain an expert adjustment performed by the instructor on behalf of students. Meta-literacy begins when students also learn to perceive the board, test the assignment, compare positions, and participate in redesign. The StudioLab extension is therefore modest but consequential: turn Vee's adaptable teaching framework into a shared practice of reading and changing the apparatus.

Stress-Test, Redesign, Circulate

The newsletter's stress-testing procedure is the hinge. Break an assignment into component tasks; state the learning goal of each task; identify what AI can now perform; test the vulnerable steps; decide what students should do unaided, what AI may assist, and what evidence will make learning perceptible. This is more than defensive assessment design. The assignment becomes executable writing about executable writing: a set of instructions whose declared purpose can be compared with what students and models can actually do.

Publication extends the test. A redesigned assignment posted to a newsletter enters comments, workshops, classrooms, podcasts, book groups, and later revisions. Teachers adapt it under different constraints; students expose effects the designer did not predict; collaborators reorganize the lesson for other disciplines. The redesign is therefore not complete when the prompt is rewritten. It acquires a public afterlife, and that afterlife becomes evidence for another round.

Substack as Perfumative Infrastructure

This recursive afterlife is what the suite calls perfumative. A post performs an explanation or recommendation, but it also releases traces: replies, citations, workshop variants, policies, copied language, refusals, examples, and institutional decisions. Some traces remain close to the author; others drift beyond attribution or return in altered form. Substack intensifies this condition because it combines essay, archive, subscription, comment thread, forwarding mechanism, and publication metric. It is simultaneously writing surface and circulation apparatus.

The feed therefore makes visible a transition from literacy to metaplay. Coding literacy teaches us to read writing as infrastructure. AI-aware teaching asks how that infrastructure reorganizes learning. Substack shows the pedagogical argument circulating through another infrastructure whose audience, cadence,

ownership, metrics, and commercial relations also shape what can be said and who encounters it. The metaplayer does not denounce that board from nowhere. The metaplayer asks how Norton's reach, Vee's voice, teacher response, student experience, and platform circulation distribute agency—and how the arrangement might be redesigned.

What This Adds to “Annette Vee as Metaplayer”

The newsletter/book pair supplies a concrete answer to the question of scale. Metaplay is not only what happens inside a classroom activity. It also happens when an idea moves among media and institutions: scholarly research becomes a public post; the post becomes a workshop; workshop questions become a guide; the guide becomes a reading group; reader adaptations return as new cases. No single artifact controls the sequence. The pedagogy develops through linked acts of publication, adoption, response, and replay.

Vee is thus a metaplayer not because she stands above the game, but because she moves among its positions: researcher, writing teacher, program administrator, newsletter author, Norton collaborator, co-author, workshop facilitator, and listener to students and colleagues. The important object is not her mastery of every position. It is the designed circulation among them. With Norton and for teachers names that circulation in its productive ambiguity.

A Situated Intellectual Relation: Madison → Pitt

The relation staged in this volume is not constructed solely from books, posts, and conceptual resemblance. Annette Vee completed her PhD in English, in composition and rhetoric, at the University of Wisconsin–Madison. She later brought Jon McKenzie to the University of Pittsburgh to speak—an encounter recalled here by McKenzie himself. Vee thus knows his work not only as a citation or distant archive, but through an academic relation joining Madison, Pitt, performance studies, writing studies, and technology pedagogy.

That history changes the status of the Vee–McKenzie conjunction. It does not license the claim that Vee endorses the StudioLab terms perfumative, metaplayer, D0–D4, or Heideggerian Engineering, and this volume does not speak for her. It does establish a situated basis for the cross-reading. The argument develops inside an existing circuit of study, invitation, conversation, and institutional passage: Madison → Pitt → AIR/StudioLab. The arrow matters because the relation itself is a kind of metaplay: concepts travel with people, are restaged in new institutions, and return altered by new technical conditions.

Sources for Appendix A

[Vee, “Introducing AI & How We Teach Writing”](#) — the newsletter's account of its Norton collaboration, audience, cadence, and classroom orientation.

[Vee, “What Is Critical AI Literacy?”](#) — functional, rhetorical, and ethical AI literacy joined to encounter, use, evaluation, and resistance.

[Vee, “AI-Aware Teaching Examples”](#) — stress-testing assignments by separating tasks, goals, vulnerabilities, and choices to resist or embrace AI.

[Vee, “The Norton Guide to AI-Aware Teaching Is Out Now!”](#) — the co-authored guide's pedagogy-first rationale, disciplinary range, and collaboration story.

[W. W. Norton, The Norton Guide to AI-Aware Teaching](#) — official overview, authorship, organization, learning-outcome focus, and publication infrastructure.

[Vee, curriculum vitae \(Pitt-hosted\)](#) — education record: PhD in English, Program in Composition and Rhetoric, University of Wisconsin–Madison, 2010; institutional bridge into the Pitt relation.

APPENDIX B. V / VEE / VOL. / VOL / GLAS

Flight, Theft, Wings, L + L, and the Letter-Game of Meta-Literacy

VEE → V → VOL. → VOL → FLIGHT / THEFT / WINGS

A Note on Method

This appendix is not an etymology of Annette Vee's name, and it does not claim that Vee intentionally encodes Derrida's *Glas*. It is a StudioLab reading machine: a deliberately excessive passage from a proper name to a sound, a letter, an abbreviation, a French word, a book form, and a pedagogical operation. Its warrant is performative rather than biographical. We ask what the mark V can do when placed inside the conceptual atmosphere of the suite.

The procedure follows *Glas* more than it explains *Glas*. Derrida's transmedia textbook makes typography, sound, translation, quotation, proper names, and dictionary definitions interfere with linear argument. A word does not sit still as a transparent carrier of one concept; it migrates across languages and material forms. Reading becomes the work of choosing paths, hearing echoes, and tracking what each path excludes. The appendix adopts that license while marking its own inventions.

Vee Becomes V

Vee is already a metalinguistic proper name: spoken aloud, it names a letter. The scholar's surname can become a minimal graphic instruction—V—and the letter can return to the name without exhausting the person. That oscillation matters for a theory of executable writing. A name is address, signature, attribution, search term, citation key, and data field; V is also a character that can be stored, sorted, parsed, displayed, or treated as an operator in a notation system.

The move from Vee to V therefore stages the difference between person and position. Vee names an author with a specific body of work. V names a slot that other readers can occupy: a fork, an opening, a descent and ascent, two strokes meeting at a point. The reduction is dangerous if it erases the person, but productive if it remains reversible. Meta-literacy lives in that reversibility—the ability to use a mark without forgetting the relations, labor, and exclusions that let it function.

Vol: Flight, Theft, and Two Wings

French gives *vol* at least two decisive semantic fields: flight and theft. The same visible word can name taking to the air or taking what belongs to another. The lexical field of flight also includes wings: historical French uses *vol* for the wingspan or wings of a bird, and heraldry uses a *vol* for a pair of wings, with *demi-vol* naming one wing. Flight is therefore already doubled. One wing is not yet the full *vol*; a second wing makes lift possible, something our AIR volumes disseminate.

This flight/theft hinge is exact enough to be dangerous for generative AI. The system promises lift: speed, variation, access, fluency, release from drudgery, a draft where there was none. But the lift is inseparable from taking: scraped text, appropriated style, extracted labor, captured attention, enclosed data, displaced credit, and the quiet conversion of common cultural memory into proprietary capability. The point is not that every use is simply theft. It is that the same operation can appear as flight from one position and taking from another.

Now add the bibliographic abbreviation *vol.* A volume is a bounded book-object, while French *vol* escapes the ground and crosses a boundary. *AIR Volume X thus becomes a compact joke with theoretical force: AIR names the medium of flight; Volume becomes vol.; vol names flight, theft, and a pair of wings.* The finished collection announces its own refusal to remain finished. This volume wants to take flight, but meta-literacy asks what it takes in order to fly.

Glas: Two Columns That Make the Reader Move

Glas is built in two facing columns, conventionally associated with Hegel on the left and Jean Genet on the right, interrupted by insertions and typographic changes. The layout does not merely decorate an argument. It breaks the sovereignty of a single reading order. A reader must decide where to enter, when to cross, what to postpone, and how to hear one column inside the other. The page becomes a board on which reading positions are produced. One shuttles bad mittens *sans* net.

The columns/*colons* also make writing architectural and material. They resemble philosophical pillars, carved surfaces, parallel legs, or adjacent channels; their interruptions expose the labor of inscription that ordinary pages try to make transparent. Katie Lloyd Thomas describes how *Glas* foregrounds the materiality of text by forcing readers to navigate notched and incised columns rather than follow an invisible conduit of meaning. This is close to the suite's distinction between interface fluency and meta-literacy: the reader must perceive the construction of the reading surface itself.

Glas also works through sound and proper-name interference. Eagle and Hegel, Genet and *genêt*, *glas* and its neighboring sounds refuse clean separation between concept, name, and material mark. The relevance to Vee is methodological. Vee/V is not a secret key hidden in her work. It is an opportunity to make the proper name vibrate against a graphic sign, just as a model's smooth answer can be made to vibrate against the data, interfaces, and institutions it usually conceals. Shuttle kooks.

L + L = V? A Graphological Proposition

L + L → TWO COLUMNS → TWO HALF-WINGS → V

Here the appendix becomes openly speculative. Take L as line and limit, leg and column: a vertical support with a foot. Put L beside L and we have a minimal two-column architecture, L + L. Angle the two uprights toward one another and their meeting produces V. Reverse the operation and V unfolds into two lines, two columns, two half-wings.

This is not a historical derivation of the letter. It is a diagrammatic move. L + L s/plays us here/now.

The plus sign matters because the second L is not a duplicate that merely completes the first. In *Glas*, the two columns interfere; in a wing pair, the two sides coordinate while producing different pressures; in AI-aware teaching, human and machine acts do not become equivalent because they appear in one workflow. L + L should therefore be read as relation plus difference. The joint is where lift is generated and where asymmetry can be hidden.

Seen this way, V is not a stable logo but a hinge. It can point downward as descent, upward as an opening, sideways as an arrowhead, or outward as wings. A metaplayer reads the hinge before treating it as an identity. Who supplied each stroke? What joins them? Which side bears weight? What appears only because the joint has been cropped out of view? *The letter becomes a tiny Gestell: a frame that orders relations into a simple, portable mark : Ges+L takes off—*

V-O-L: A Miniature Metaplay

The word *vol* can itself be played as a three-position diagram. V forks and joins; O loops, circulates, encloses, or opens an aperture; L stands, turns, and establishes an edge. Read dynamically: the fork enters a loop and exits at a limit. Read institutionally: alternatives are offered, recirculated through a platform, and stabilized as policy. Read pedagogically: a learner chooses, traces the circulation of the choice, then redesigns the boundary that made the choice legible.

Because this is metaplay, no spelling becomes final doctrine. The purpose of V-O-L is to train movement among representations: phonetic, graphic, lexical, bibliographic, architectural, computational, and institutional. Each representation reveals something and steals something from view. The reader's task is to make the loss traceable and the gain contestable.

Pedagogical Protocol: Make It Fly, Trace What It Takes, Flail Fast

Give participants a short AI-generated passage and ask them to perform two operations. First, make it fly: improve its mobility by changing genre, audience, medium, language, or level of abstraction. Second, trace what the flight takes: identify borrowed language, missing sources, displaced labor, hidden compute, policy assumptions, and forms of attention the transformation consumes. The same revision must be read from the position of lift and the position of taking.

Then split the page into two columns. In the left column, place the declared purpose and the human/model workflow. In the right, place the unplanned effects, absent voices, and redistributed credit. Finally fold the two columns into a V by writing one redesigned rule at their point of contact. The exercise converts wordplay into meta-literacy: flight becomes accountable only when its thefts, transfers, and dependencies can be sensed and redesigned.

From Metaplayer to Meta-Flier

Vee as metaplayer now acquires a winged double: the meta-flier. The meta-flier does not simply rise above the board, since altitude can reproduce the fantasy of sovereign oversight. It moves between ground and air, close reading and overview, one column and the other, the bounded volume and its circulating traces. It knows that every view from above is supported by an apparatus and that every escape route has a logistics.

The final formula is therefore not Vee equals *vol*. It is Vee gives us a reason to play V through *vol* inside an AIR volume after Glas. The play sharpens the suite's central demand: do not ask only whether AI helps writing take flight. Ask what is taken, whose wings are clipped, how the lift is engineered, and whether the people carried by the apparatus can alter its design.

Sources for Appendix B

[CNRTL, "Vol"](#) — historical senses of flight, figurative departure, wingspan, and the heraldic pair of wings or demi-vol.

[Jacques Derrida, Glas](#) — the flight/theft wordplay and the book's two-column practice of nonlinear, materially conspicuous reading.

[Katie Lloyd Thomas, "Specifications: Writing Materials in Architecture and Philosophy"](#) — an account of Glas as a carved, columnar writing surface that makes textual materiality perceptible.

[John Llewelyn, "Glasnostalgia"](#) — the two-column interplay of Hegel, Genet, flowers, animals, pronunciation, signature, and the resistance of the mark.

APPENDIX C. PYNCHON'S V.: THE LETTER AS QUEST-MACHINE

Stencil, Profane, Archive, Inanimation, and the Metaplay of the Variable

V. = WOMAN / PLACE / INITIAL / ARCHIVE / VARIABLE /
VANISHING POINT

Why V. Requires Another Appendix

The previous appendix moves from Vee to V, vol., vol, wings, theft, and *Glas*. Pynchon makes that chain less ornamental and more dangerous. In *V.*, a single letter does not simply accumulate meanings; it organizes a search. Herbert Stencil gathers documents, memories, names, places, bodies, and historical crises under the sign V, while Benny Profane drifts through a world in which people and things repeatedly exchange properties. The letter is therefore not a symbol waiting to be decoded once. It is a machine for recruiting evidence, arranging coincidence, and keeping interpretation in motion.

That machine belongs inside a theory of AI meta-literacy. Generative systems offer users an abundance of plausible fragments and invite them to synthesize patterns across them. The danger is not only false information. It is the Stencil problem: a sufficiently energetic search can make an archive appear to confirm the object it was organized to find. Pynchon supplies the suite with a literary model of pattern hunger—an intelligence that can be brilliant, mobile, and generative while remaining unable to prove whether its central object exists.

The Period.s After the Letter

Pynchon's title is not merely V but *V.* The period makes the mark look like an abbreviation whose missing expansion has become the novel's engine. It promises that the letter stands for something while withholding the rule that would settle what that something is. The dot can be read as closure, but it also multiplies suspense: sentence end, decimal point, puncture, destination, or the small remainder that prevents V from becoming a transparent name. The title says both stop and continue: it *s/plays*....

This punctuation sharpens the vol./vol hinge already at work in the AIR volume. Vol. appears to abbreviate a stable bibliographic object; *vol* releases flight and theft. *V.* intensifies the trick by turning abbreviation into quest.ion. A reader sees an incomplete instruction and begins supplying expansions. That activity is performative: the mark produces investigators, dossiers, hypotheses, routes, and disputes. *V.* does not need to denote one stable entity in order to act. Its indeterminacy is precisely what allows it to organize a field.

Two Arms: Stencil and Profane

The novel's two major trajectories can be drawn as the arms of its title letter. Stencil searches, reconstructs, and narrativizes. Profane yo-yos, wanders, reacts, and repeatedly finds himself handled by systems he neither masters nor fully resists. One converts movement into pattern; the other undergoes movement without converting it into a durable project. Their paths intersect, but neither supplies the sovereign view that would resolve the book. The point of the V is not a triumphant convergence. It is a joint where two incomplete modes of inhabiting modernity become visible together.

These are also two familiar positions in AI use. Stencil resembles the user who prompts repeatedly, gathers outputs, detects motifs, and constructs an increasingly coherent theory from model traces. Profane resembles the user who moves from interface to interface, accepting recommendations and generated language as the environment's weather. Critical literacy usually warns Profane to become more active. Pynchon adds the inverse warning: activity itself can become compulsive pattern production. Meta-literacy must move between the two arms—able to investigate without forcing closure and able to drift without surrendering agency.

V. as Archive and Synthetic Dossier

The most productive scholarly account of Pynchon's letter treats V. as an archive. Stencil inherits fragments and builds a dossier whose coherence is never guaranteed by the materials themselves. Names, bodies, places, dates, and geopolitical disturbances become legible as traces because he has already supplied the search term that gathers them. The archive preserves debris, but it also redistributes that debris through selection, description, order, and omission. V. is thus not hidden intact behind the documents. V. is partly produced by the procedure that claims merely to recover her.

A generative model is not an archive in the conventional sense, yet it creates a comparable epistemic temptation. Training materials are transformed into statistical dispositions, and a prompt elicits a synthetic dossier rather than retrieving an untouched original. The answer can feel like the missing center because dispersed traces have been compressed into a fluent surface. Pynchon helps name the proper response: treat the output as a Stencil—an arrangement that reveals relations while also exposing the desire that organized them. Ask what search term made these fragments neighbors, what alternatives were not recruited, and what would count as disconfirming evidence.

The Violence of Making Woman into a Sign

Pynchon's V. repeatedly takes shape through female figures, fetishized bodies, prostheses, aliases, and scenes of historical violence. That structure cannot be borrowed innocently for a play on Annette Vee's surname. The male investigator's desire for coherence turns women into evidence for his quest, and the novel's traffic between animate person and inanimate object makes that conversion part of its horror. A metaplay reading must therefore examine not only the flexibility of the letter but the violence by which a person is reduced to a mobile sign.

This warning retroactively disciplines Appendix B. Vee is not V.; her name is not a cryptogram whose truth the suite can possess. The reversibility proposed there—Vee becoming V and V returning to Vee—must preserve the asymmetry between a living scholar and a letter-game authored around her. The play is warranted only when the person can exceed, refuse, or redesign the positions assigned by the apparatus. Otherwise meta-literacy repeats Stencil's mistake at another scale: it celebrates its own ingenuity while converting someone else into the object that keeps the game alive.

The Inanimate, Gestell, and Generative AI

Across Pynchon's novel, the border between animate and inanimate erodes. Bodies acquire prosthetic and mechanical features; objects solicit attachment; people behave like repeatable mechanisms; historical systems make human beings available as components. Heidegger's Gestell offers a related vocabulary of ordering, but Pynchon supplies the grotesque phenomenology of that ordering. Enframing is not only an abstract disclosure of beings as standing reserve. It is felt as the comic and terrifying uncertainty over whether a person is using a machine, loving an object, imitating a mechanism, or becoming one.

Generative AI intensifies this uncertainty because language associated with human attention, judgment, and address returns from an inanimate computational apparatus with the texture of a responsive voice. The correct lesson is neither that the model is secretly a person nor that its outputs are inert and inconsequential. Perfumative systems release effects across that distinction. Their language enters classrooms, evaluations, relationships, and institutional decisions. A Pynchonian meta-literacy tracks the crossings: where animation is projected onto the system, where people are rendered as data or workflow components, and where responsibility disappears into the exchange.

From Variable to Metaplayer

V also behaves like a variable. It can hold a person, place, event, body part, historical tendency, or nothing determinate at all. But unlike a neutral algebraic placeholder, Pynchon's variable changes the behavior of everyone who tries to solve it. It is operational before it is resolved. Stencil becomes Stencil through the pursuit; documents become clues by entering his dossier; readers become co-investigators when they test one candidate against another. The variable scripts positions on the board.

That is the decisive connection to metaplay. The player asks which candidate fills V. The critic asks why Stencil needs V. The metaplayer examines how the variable, the dossier, the investigator, the archive, and the reader recursively produce one another. Applied to AI, this means that a prompt is not merely a request for content. It establishes a variable, recruits a corpus-shaped response, positions a user as evaluator, and may install the output as evidence for the frame that generated it. Meta-literacy makes that circuit available for redesign.

A Bigger Boat: The Volume as Vessel

Fishing for great white sharks in multi-colored cloaks, we do need a bigger boat because V will not fit inside a single lexical skiff or kraft. The letter now carries Vee, vol., volume, wings, theft, *Glas*, Pynchon, variable, and vanishing point. But scale alone is not the answer. A larger vessel can carry more cargo while hiding more of its manifest, labor, ballast, ownership, and exclusions. Our AIR volume should therefore behave less like a container that possesses its contents than like a vessel whose loading procedures remain visible and revisable.

Pynchon's naval drifters and Mediterranean crossings add a maritime counterimage to the meta-flier. Flight promises overview; the boat remains inside the medium that carries and threatens it. The metaplayer sometimes needs altitude, but also immersion, drag, weather, and dependence on a crew. The passage from meta-flier to meta-mariner prevents metaplay from becoming a fantasy of looking down on everyone else's game. One navigates from within infrastructures whose currents cannot be mastered from a single position of a drunken sailor.

StudioLab Protocol: Stencil / Profane / V.

Give participants one AI-generated answer to a contested question and assign three positions.

Stencil marks every recurring motif, corroborating detail, source-shaped gesture, and apparent clue; the aim is to build the strongest possible account of the answer's hidden V.

Profane moves through the same answer without solving it, recording nudges, defaults, frictionless transitions, and moments when the interface decides what deserves attention.

V. remains an empty heading under which participants place several rival explanations rather than one conclusion.

The group then metaplays the three positions. Which observations became evidence only because Stencil's frame selected them? What did Profane notice precisely because no grand pattern organized his movement? Which institutional actors—teacher, student, platform, publisher, administrator, dataset worker—were absent from both accounts? The final move is not to identify the true V. It is to rewrite the prompt, evidence rule, or classroom protocol so that competing patterns can be tested, provenance can be traced, and the people turned into signs can answer back.

Why Pynchon Changes the Vee Suite

Pynchon adds a missing negative capability to the progression from coding literacy to metaplay. Executable writing encourages us to ask what language does. Revealing and grammatization show how technical systems order what can appear. Performativity and perfumativity follow acts into their distributed effects. V. adds the question of interpretive overproduction: when does the apparatus of inquiry manufacture the unity it claims to discover? *Without that question, metaplay can become merely a more elaborate confidence in pattern recognition.*

The play splays, divides and sorts and stops. *and +L → V. + L → vol → ... !?*

The revised chain is therefore Vee → coding literacy → executable writing → revealing → grammatization → perfumativity → AI-aware teaching → metaplay → V. The final period does not close the sequence. It marks a disciplined suspension. A meta-literate reader can build patterns, test them, inhabit them, and redesign their rules while keeping open the possibility that the desired center is an effect of the search. Pynchon's V. is not the answer to the suite. It is the device that prevents the suite from confusing a powerful reading machine with the truth!

Sources for Appendix C

[Thomas Pynchon, V. \(publisher overview\)](#) — the intersecting trajectories of Benny Profane, Herbert Stencil, the Whole Sick Crew, and the unknown V.

[Meredith K. Hegg, "V. as Archive"](#) — V. as archival character, marker, repository of multiple subjectivities, and point of narrative reconstruction.

[Richard Poirier, "Cook's Tour"](#) — a contemporary account of the cryptographic title, the novel's proliferating V-referents, and its animate/inanimate disturbance.

APPENDIX D.

CRYPTO-HAUNTO-HYSTERICAL FOAMATIONS

V as the Bifurcating Letter of Dividuation

V↓ = MANY TRACES → PROFILE /

V↑ = PROFILE → OPERABLE FRAGMENTS

V as a Reversible Operator

V is usually read from its two upper arms toward the point below: two lines converge, alternatives narrow, scattered traces become one mark. Reverse the direction and the same letter bifurcates: a point divides into branches, a unit becomes a pair, and an apparent identity distributes itself across divergent paths. The letter therefore diagrams two operations at once. Downward, it funnels; upward, it forks. Its metaplay lies in making the direction of reading available for choice.

Dividuation occurs in both directions. A platform decomposes a person, text, or event into tokens, scores, embeddings, clicks, roles, and probabilities; it can then recombine those fragments into a profile, recommendation, risk category, or generated response. The funnel does not recover the original individual, and the fork does not merely reveal preexisting parts. Each operation produces the units it claims to process. V is the graphic hinge between decomposition and actionable recomposition.

Cryptoformation

A cryptoformation organizes a field around something encoded, withheld, or accessible only through a key. Pynchon's V. is exemplary: a minimal sign makes names, locations, documents, bodies, and coincidences look like ciphertext awaiting a master reading. In AI systems, the hidden center may be a model's parameters, an undisclosed training corpus, a ranking rule, or the institutional objective embedded in an interface. Cryptoformation does not mean that a secret necessarily exists. It names the formation produced when actors behave as if visible traces must decode an absent center.

Hauntoformation

A hauntoformation is organized by what is absent yet still effective. Archived language, suppressed histories, abandoned platforms, dead metaphors, prior versions, and excluded voices return as traces that disturb the present without becoming fully present themselves. The temporal logic is neither simple memory nor retrieval. What returns has been transformed by the apparatus of return. In generative AI, a phrase can arrive as an uncited echo of training, a style can persist without its maker, and a discarded institutional decision can survive as a default long after anyone remembers choosing it.

Hysteroformation

A hysteroformation begins from division and addresses a presumed knower: tell me what I am, explain what this means, name the cause, complete the pattern. The demand produces knowledge, but no answer resolves the division that generated it. Prompting can occupy this position when the user repeatedly asks the model to supply the master signifier that would stabilize an identity, diagnosis, history, or interpretation. The model's fluency can intensify the circuit by making every provisional answer feel like a key while also generating the next question.

Foamation

Foamation transforms formation from a solid structure into a plurality of adjacent bubbles. Sloterdijk's foam offers no single globe that contains everyone and no isolated atom that exists alone. Each cell maintains an interior through walls it shares with neighbors. The wall connects and separates at once; pressure in one cell reshapes the others. Foamation therefore names distributed co-formation through selective permeability, local atmosphere, shared infrastructure, and unequal exposure.

Digital life is foamic before it is universal. A classroom, newsletter, platform, publisher, model, policy office, dataset, and discussion thread can share walls without merging into one public. Language, metrics, permissions, and anxieties pass across those walls at different rates. Some bubbles are climate-controlled; others absorb the heat and waste. Meta-literacy asks not only who belongs to the network but how its membranes filter movement, protection, credit, risk, and the right to alter the shared boundary.

The Crypto-Haunto-Hysterical Composite

Cryptoformation, hauntoformation, and hysteroformation can lock into a single composite. Something missing returns as a trace; the trace appears coded; the divided subject demands that an authority decode it. The resulting knowledge does not end the search because the absent center is also what keeps the apparatus moving. Pynchon's Stencil turns fragments into V; the AI user can similarly turn outputs into clues about an intelligence, corpus, self, conspiracy, or future that the outputs can never simply disclose.

Crypto-haunto-hysterical is a StudioLab construction, not a term attributed to Pynchon, Derrida, Lacan, Deleuze, or Sloterdijk. Its value is diagnostic. It identifies a recursive formation in which concealment, return, and demand amplify one another: secrecy makes the trace magnetic; haunting makes it persistent; hysteria makes it productive of more queries and more knowledge. The metaplayer interrupts the loop by asking what would count as a non-clue, whose absence is being used, and which authority benefits from appearing to hold the key.

Dividuation: One Becomes Many, Many Become One

Deleuze's dividual names the person rendered divisible within continuous systems of control: samples, data, markets, banks, credentials, probabilities. Dividuation does not simply destroy individuality. It makes fragments operational in different settings. The same learner becomes a writing score, attendance event, risk flag, prompt history, demographic row, and predicted outcome. No fragment is the whole person, yet each may trigger a consequential decision.

V maps this double movement with unusual economy. Read upward, the point called person or event branches into operable traces. Read downward, heterogeneous traces converge into a profile that can be acted upon as if it were one. The critical question is not merely whether the profile is accurate. It is who chooses the branches, who controls the point of convergence, which fragments remain portable across bubbles, and whether the dividuated subject can inspect, contest, or redesign the operation.

Four Couplings

CRYPTO × HAUNTO → THE ENCRYPTED RETURN

A trace returns without its origin as a cipher.
The archive becomes a field of possible keys;
AI output becomes evidence
of a hidden corpus or intention.
Metaplay tests alternative keys and
preserves chances that no single plaintext
lies behind the trace.

HAUNTO × HYSTERO → THE DEMANDING G/HOST

What has not been settled returns
as a question addressed to a knower.
The model, teacher, institution, or theory
is asked to name the missing truth.
Metaplay hears the demand without
installing the respondent as infallible master.

HYSTERO × FOAM → CONTAGIOUS QUESTIONING

A question moves across adjacent bubbles
and changes pressure as it travels.
Student anxiety becomes teacher policy,
policy becomes platform behavior,
platform behavior becomes publisher guidance,
and guidance returns as classroom expectation.
Metaplay traces shared walls circulating demand.

FOAM × CRYPTO → SELECTIVE PERMEABILITY

Every bubble has keys, interfaces, thresholds.
Some traces cross easily while other
actors and histories remain blocked, sealed out.
Metaplay treats access control, privacy,
provenance, and visibility as mem-
brain design rather than after-the-fact ethics.

AI-Aware Teaching: Read the Walls, Keys, and Arrows

An AI-aware exercise can begin with one generated response and four passes. The crypto pass marks clues and asks what hidden center they seem to promise. The haunto pass marks echoes, absences, inherited phrases, and unfinished histories. The hystero pass identifies the divided subject, the authority being addressed, and the knowledge the demand produces. The foam pass maps the institutional cells and shared walls through which the response acquires force.

Participants then draw V twice. In the upward version, they place the person or event at the point and label the traces into which the apparatus divides it. In the downward version, they show how those traces are condensed into an actionable profile, answer, or decision. The redesign must alter at least one key, wall, branch, or convergence rule. That requirement turns critique into metaplay: the formation is not only interpreted but made available for collective modification.

Map Legend: Four Questions

Crypto asks: what made this fragment count as evidence? Haunto asks: what unresolved past is programming the present? Hystero asks: who is being asked to tell the subject what it is? Foam asks: which walls connect, filter, protect, or exclude? V adds the transversal question: are we watching one become many, many become one, or both operations recursively feeding each other?

Sources for Appendix D

[Gilles Deleuze, "Postscript on the Societies of Control"](#) — individuals, coded access, continuous modulation, and persons rendered as samples, data, markets, or banks.

[Jacques Derrida, Specters of Marx](#) — spectral return, disjointed time, inheritance, and the unfinished efficacy of what is no longer simply present.

[Bruce Fink, "The Master Signifier and the Four Discourses"](#) — the divided subject, the master signifier, and knowledge production in the hysteric's discourse.

[Peter Sloterdijk, Foams](#) — plural spherology, capsules, islands, hothouses, and modern community as a multiplicity of adjacent interiors.

[Meredith K. Hegg, "V. as Archive"](#) — V. as marker, archival character, repository of multiple subjectivities, and engine of narrative reconstruction.

MeTanoia Playground

StudioLab

"PLAY is the lab where we unlearn to learn again."

Care is a route.

We wander to make cure.

maps hold memory

THE GROUND REMEMBERS.

SONG KEEPS US ALIGNED.

healing is cosmic work.

WE WITNESS.
WE REMEMBER.
WE IMAGINE.
WE PLAY.

Labyrinth

nothing is linear here.
get lost.
find more.

Serpentine

follow the curve.
trust the uncurling.

The Fall / Cliff

risk reminds us we are alive.

border between play & thought.

RETURN is a DREAM WE KEEP.

every path can hold the others

Rock Garden

step gently.
listen to the stone.

String Fingers

climb.
pluck.
become vibration.

Care Station

rest. repair.
recharge.

Witness Bench

sit. watch.
hold the story.

Dorita's Doric Doorway

pass through.
leave different.
return wiser.

Cosmic Egg

crack open.
enter the unknown.

Cloud Machine

make weather.
make wonder.

StoryX Path

every step
every story.
every story a step.

Toryx Switch

choose.
switch.
transform the map.

Haunted House

enter if you dare.
leave what you carry.

play is rehearsal for new worlds

WE BUILD TOMORROW WITH TODAY'S PLAY.

Pablo's Frame Shop

reframe.
reshape.
reimagine.

shadows teach us how to see.

Future-Present Space

where we practice what's possible.

beauty is our act of defiance.

Children paint tomorrow.

The Playground

A StudioLab Performance Cosmography

MAP KEY

- Path / Route
- Portal / Threshold
- Witness Point
- Care Station

This map is not fixed.
It changes with us.
Draw your own path.
Add your own place.
The playground is alive.



beyng.com
14BC Gallery NYC

performingspace.org

