

TEACHING NOTE

Producing the Anti-ESG Ecosystem: Designing and Managing a Hybrid Intelligent Team

Synopsis

This case documents how Robert G. Eccles, a retired Harvard Business School professor, produced the book “The Anti-ESG Ecosystem: Implications for U.S. Climate Policy” using a structured Hybrid Intelligent Team (HIT)—a coordinated ensemble of five AI systems, each assigned a distinct functional role. The project extended over several months and required sustained analytical work: mapping organizations, funding flows, and political strategies behind the anti-ESG backlash in the United States and assessing the implications for climate policy.

The case is not an account of AI-assisted writing in any generic sense. It is an account of process design under uncertainty. Eccles did not know, at the outset, how a multi-agent AI workflow would perform under conditions of sustained, complex intellectual work. What he discovered was that role clarity was a necessary but not sufficient condition for reliable execution. The systems could misinterpret instructions, compress arguments, drift in tone, and rewrite when editing had been requested—not out of incapacity, but because they interpreted instructions rather than executing them.

The response was not better prompting. It was process redesign—decomposing tasks into bounded steps, differentiating conversational environments by function (drafting, revision, evidence integration, and methodological reflection), building a locking discipline to fix completed work against unnecessary revision, and managing human oversight not as a fallback but as the primary mechanism of quality control.

This note was prepared by Professor Robert G. Eccles, Claude (Primary Drafter and Synthesize), Clean ChatGPT (Independent Reader), Dorothy ChatGPT (Principal Collaborating Agent), Gemini Pro (Analytical Critic & Adversarial Reviewer), and Perplexity (Researcher and Evidence Specialist) for aiding classroom discussion in the use of “Producing the Anti-ESG Ecosystem: Designing and Managing a Hybrid Intelligent Team. HLS No. 001.

The case concludes with a problem that the workflow could not fully solve. The tacit knowledge embedded in the process—the specific judgments about timing, sequencing, and evaluation standards—could not be reliably recovered from the conversation record alone. Whether that knowledge could be made explicit enough to be transferable was the question that remained open.

The production of the case itself followed the same method. Dorothy (an identity-specific instance of ChatGPT) was the Principal Collaborating Agent (PCA). Claude was the main case writer. Gemini Pro served as an adversarial reviewer of the logic and argument of the case. Clean ChatGPT (memory off, so no Dorothy) served as an independent editor providing feedback to produce the final version of the case. The case is thus both a description of a HIT-based production process and an instance of one. The production of this Teaching Note itself serves as a further instance of the same Hybrid Intelligent Team method with the addition of Perplexity who performed the necessary research regarding relevant academic literature.

Teaching Objectives

By the end of class, students should be able to:

- (1) Distinguish between role design and process design in human–AI collaboration and explain why the latter is the more consequential variable in sustained complex work.
- (2) Evaluate the tradeoffs between control and scalability in AI-assisted workflows and identify the conditions under which tighter human oversight is not an inefficiency but a structural requirement.
- (3) Analyze the limits of prompt-based systems: why more explicit instructions improve outputs at the margin but do not resolve the underlying dynamic of interpretation rather than execution.
- (4) Recognize tacit knowledge as a design variable: diagnose where tacit knowledge is embedded in a workflow, assess the conditions under which it can be made explicit, and understand the residual limits of that effort.
- (5) Apply task decomposition principles to an AI-assisted workflow, including the use of functionally differentiated environments, locking protocols, and independent verification channels.
- (6) Assess the transferability of a documented HIT workflow and identify what an account of process can and cannot convey to a practitioner operating in a different context.

Analytical Framework and Key Themes

Six themes organize the analytical architecture of this case. Each connects directly to the case evidence; several are in productive tension with one another.

Theme 1: Role Design vs. Process Design

The initial architecture of the HIT—Claude for drafting, Gemini for adversarial challenge, Perplexity for verification, Dorothy for coordination, and the clean ChatGPT instance for independent assessment—represents role design: defining what each system does and why. Role design is a necessary starting condition, but the case demonstrates that it is not sufficient.

The failures that emerged early in the project—compression, tone drift, rewriting when editing was requested—were not failures of role assignment. Claude did not malfunction. It interpreted. The gap between instruction and execution is not a communication problem; it is a structural feature of systems that process language in terms of meaning rather than logic. When Eccles encountered this gap, the productive response was not to keep specifying roles more precisely. It was to redesign the process: decompose tasks, differentiate environments, build locking disciplines, and sequence human evaluation at each stage.

Conceptual grounding: Galbraith's organizational design principle—that information-processing requirements should drive structural choices—applies directly. Simon's architecture of complexity provides the decomposition logic: a system whose failures are bounded is more manageable than one whose failures propagate. Baldwin and Clark's modularity framework illuminates the room structure: each room is a module with defined inputs, outputs, and independence from other modules. See Jay Galbraith, *Designing Organizations* (1973); Herbert Simon, "The Architecture of Complexity" (1962); Carliss Baldwin and Kim Clark, *Design Rules* (2000).

Theme 2: Task Decomposition and Error Containment

The most consequential structural change Eccles made was decomposition: breaking large, multi-specification prompts into sequential single-purpose steps. The outline-before-draft protocol is the clearest instance. Asking for an outline before a draft is not simply a writing technique; it is an error-containment strategy. If the outline is wrong, the error is visible and correctable at the outline stage, before it propagates into a full draft. The principle—no task should be large enough that its failure could damage more than a single bounded unit of work—applies directly to any system where outputs cannot be fully verified before they are integrated.

The room structure embodies the same logic at a higher level. Drafting, revision, evidence integration, and methodological reflection are not just different activities; they are

modules with different failure modes, different evaluation criteria, and different standards for output quality. Housing them in separate environments prevents context saturation from one module contaminating another and makes the provenance of any given output traceable.

Conceptual grounding: Simon's notion of near-decomposability—that complex systems are more robust when subsystems interact weakly—provides the theoretical foundation. Eccles's room structure is a practical instantiation of near-decomposability in a knowledge-production context.

Theme 3: Tacit Knowledge and Its Limits

The case concludes with a problem that is, in a strict sense, unsolvable. The tacit knowledge embedded in Eccles's workflow—the specific judgments about timing, the precise formulations that succeeded where others had failed, the evaluation standards applied moment by moment—was not preserved in the conversation record and could not be reliably reconstructed from it. This is not a documentation failure. It is a feature of complex, adaptive processes. The knowledge required to execute them is partially constituted by the doing.

Polanyi's formulation—"we know more than we can tell"—is the foundational reference. The reflection room, in which Eccles used the discipline of composing a prompt to force precision about what he actually thought was happening, is a direct application of Nonaka's knowledge conversion model—tacit knowledge made temporarily explicit through articulation. But the case does not present this as fully successful. The principles that emerged from the reflection room—role separation matters more than specific tool assignments; functional differentiation matters more than specific room structure—are accurate but not quite sufficient. They describe what mattered; they do not fully convey the judgment required to apply them. The gap between description and transmission is the problem that remains.

Conceptual grounding: Michael Polanyi, *The Tacit Dimension* (1966); Ikujiro Nonaka, "A Dynamic Theory of Organizational Knowledge Creation," *Organization Science* 5(1) (1994).

Theme 4: Provenance and Revision Strategy

Eccles maintained explicit records of which system produced which section, not for attribution but as a guide to revision. Claude-drafted sections tended toward structural coherence but might lack adversarial sharpness; Dorothy-synthesized sections might carry unexamined assumptions absorbed from prior context. Knowing provenance meant knowing what kind of revision was required before reading a line of the draft.

This is a design insight that extends beyond AI workflows. In any collaborative production process, the diagnostic value of knowing who produced something—and with

what tendencies and constraints—is a precondition for intelligent revision. Eccles's documentation practice converted a record of production into a revision guide.

The locking discipline reinforces the same logic from the opposite direction. Locking a section is not just quality control; it is the establishment of a stable platform from which subsequent work can proceed without destabilizing what has already been verified. The standard for locking—argument sound, prose right, and evidence in place—is analytically demanding, and the practice of writing a note identifying inadequacies and returning the following day is a form of built-in review against the temptation to lock prematurely.

Theme 5: Human Oversight as Structural Requirement

A persistent misconception in discussions of AI-assisted work is that automation reduces the need for human judgment. The case presents the opposite finding. As the workflow became more structured, the role of human judgment intensified rather than diminished. Eccles's evaluation time ran at ten or more to one relative to generation time. The evaluation was not passive; it required sustained analytical attention to argument structure, tonal consistency, evidential grounding, and structural fit within the chapter and the book as a whole.

Brynjolfsson and McAfee's characterization of human–machine complementarity—that the relevant question is not whether machines can replace humans, but where each has comparative advantage—is directly applicable. The case shows that evaluation, sequencing, and the judgment of when output is good enough are not tasks that could be decomposed or delegated further. They required Eccles's specific knowledge of what the book was trying to do, developed over months of work, and not accessible to any AI system in the workflow.

Conceptual grounding: Erik Brynjolfsson and Andrew McAfee, *The Second Machine Age* (2014); Robert G. Eccles, "Hybrid Intelligent Teams" (SSRN Working Paper, 2026), https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5792345.

Theme 6: Self-Imposed Psychological Safety

Eccles's approach to failure—documented, analyzed, and converted into process redesign—reflects what can be understood as self-imposed psychological safety: a willingness to expose errors to scrutiny, invite challenge to one's own design decisions, and revise process without defensiveness. When Gemini identified that a central claim rested on circumstantial evidence, the response was not to defend the claim but to spend two additional sessions strengthening its evidentiary basis. When the clean ChatGPT instance returned a description of the chapter's argument that differed from the intended one, the response was to revise the introduction and reframe the opening sections.

This disposition is organizationally significant. It is not simply an intellectual virtue. In a workflow where the primary control mechanism is human evaluation, the quality of evaluation depends directly on the evaluator's willingness to see accurately rather than confirmatorily. A practitioner who is psychologically unwilling to acknowledge that an output is wrong—because acknowledging it implies that a prompt was badly written, or that a tool was misused, or that a section must be reopened—will produce a systematically lower-quality result than one who is not.

Conceptual grounding: Amy Edmondson, *The Fearless Organization* (2018). Edmondson's concept of psychological safety is typically applied to teams; its application here to the individual practitioner's relationship to their own process and its failures extends the concept in a direction directly relevant to solo knowledge-production workflows.

Sources of Difficulty

This case is analytically demanding in ways that are not immediately visible. Instructors should anticipate the following misreadings:

The Prompt-Solution Fallacy

Students whose primary experience with AI tools has been conversational will tend to locate the problems Eccles encountered in inadequate prompting, and to look for the solution in better prompts. The case is structured to frustrate this reading. Every attempt to solve the problem through more explicit instructions produced marginal improvement but did not resolve the underlying dynamic. The insight the case is designed to produce—that the problem is not communicative but structural, and that the solution is process redesign rather than prompt refinement—requires students to let go of a framework that has served them in simpler contexts.

Underestimating Human Judgment

The ten+-to-one ratio of evaluation time to generation time is not an inefficiency to be engineered away; it is a feature of any production process where output quality cannot be assessed without domain knowledge. Students who frame the case as a workflow optimization problem will look for ways to reduce the evaluation burden. The productive reframing is to ask what the evaluation work consists of, why it cannot be delegated further, and what that implies about the nature of knowledge work in a human–AI system.

Confusing Capability with Reliability

Each AI system in the HIT is capable, in the sense that it can produce outputs that are locally plausible and structurally appropriate. The failures the case documents are not failures of capability; they are failures of reliability under specific conditions: extended sessions, multi-specification prompts, bounded editing tasks. Students should be asked to distinguish

between what a system can do in a demonstration context and what it will do consistently across a complex, sustained production process.

Missing the Process Design Contribution

The room structure, the locking discipline, the decomposition protocol, and the provenance documentation are not administrative overhead. They are the intellectual contribution of the case. Students who read the case as a description of tools and their properties will miss what is analytically original: the design principles that made the tools collectively productive. The question to press is: what did Eccles design that is not a tool?

The Tacit Knowledge Misunderstanding

Students may interpret the case's conclusion—that tacit knowledge cannot be fully transmitted—as a counsel of despair. If the key knowledge is not transferable, what is the point of documenting it? The more productive reading is to ask what can be transmitted (principles at the appropriate level of abstraction), what cannot (judgment in context), and what the implications are for how practitioners should approach the design of their own workflows rather than attempting to replicate Eccles's.

Study Questions

The following questions are designed for pre-class preparation. Instructors should assign all five and expect that students will have thought carefully about the first three.

- (1) Eccles distinguishes between role design and process design. What does each mean in this context, and why does he conclude that role design was insufficient? What specific evidence from the case supports that conclusion?
- (2) Three recurring failure modes—compression, tone drift, and rewriting when editing was requested—are described in detail. What do they have in common analytically? What does that commonality imply about the nature of the problem Eccles was facing?
- (3) The workflow Eccles developed—outline before draft, functionally differentiated rooms, locking protocol, provenance documentation, independent clean-instance assessment—can be described as a set of design principles. State each principle in one sentence. Which principle do you find most analytically interesting, and why?
- (4) Eccles's evaluation time ran at roughly ten or more to one relative to generation time. Is this a problem to be solved or a feature to be accepted? What are the arguments on each side? Where do you come out?

- (5) The case concludes with the problem of tacit knowledge. What was known in the doing cannot be reliably recovered from the record. Does this constitute a failure of the case as a teaching document? What can the case transmit, and what can it not?
- (6) The workflow Eccles developed is deeply tied to his own accumulated judgment—his evaluation standards, his sense of when a room transition is timely, his criteria for locking. To what extent is this workflow transferable to another practitioner, and to what extent does it depend on capabilities that cannot be conveyed by documentation alone? What would a different person need to develop independently, and how?

Teaching Plan (90 Minutes)

Opening (0–10 minutes)

Open with a single question directed at the class: What is this case actually about? Accept three or four responses without comment. Expect a range: a case about using AI to write a book; a case about workflow design; a case about the limits of AI tools. Hold the tension between these framings without resolving it. The case is about all of them, but the level of abstraction matters for what students take away.

Transition: Tell the class that the case documents a specific workflow in detail. The analytical work of class is to determine what principles that workflow embodies and whether those principles generalize beyond the specific project and tools described.

Block 1: Diagnosing the Failures (10–30 minutes)

Call on a student to describe the three primary failure modes Eccles encountered. Press for precision. What exactly was going wrong in each case, and what did Eccles's attempts to correct them through prompt specification produce? The key move is to establish that more explicit instructions produced marginal improvement but did not solve the problem. Ask: What does that tell us about the nature of the problem?

Drive toward the distinction between execution and interpretation. The systems were not executing instructions; they were interpreting them. The gap between what was written and what was returned was not a failure of communication; it was a structural feature of the tools. Ask: If the problem is structural rather than communicative, what kind of solution is appropriate?

Block 2: The Process Design Response (30–55 minutes)

Ask students to reconstruct the process design that emerged in response to the failures. Push for specificity: name the rooms, name the locking discipline, name the decomposition protocol, and name the provenance documentation practice. For each element, ask: What problem is this solving, and what principle does it embody?

The goal of this block is to arrive at a set of design principles stated at the right level of abstraction—specific enough to be actionable, general enough to apply beyond this case. The four principles Eccles distilled are: (1) role separation over specific tool assignment; (2) functional differentiation over specific room structure; (3) fixed foundations over provisional ones; and (4) bounded tasks over large multi-specification prompts.

Introduce Simon and Baldwin and Clark here if students are not drawing on them. The room structure is a modularity solution. Each room has defined inputs, outputs, and failure modes, and failure in one room does not propagate to another. Ask: Is there a cost to this modularity? What is lost when you separate drafting from revision from evidence integration?

Block 3: Tacit Knowledge and Transferability (55–75 minutes)

Pivot to the case's conclusion. Read aloud the paragraph beginning: "The tacit knowledge embedded in the workflow was not easily recoverable from the record." Ask: What is Eccles saying here? What does it mean for this case as a teaching document?

Introduce Polanyi and Nonaka. The reflection room is an attempt to convert tacit knowledge into explicit knowledge—Nonaka's externalization process. Ask: Does it succeed? What does Eccles say about the limits of the principles he extracted?

The productive line of argument is that the case can transmit structure but not judgment; it can describe what was done and why, but it cannot fully convey the feel of knowing when a section is good enough, or when a room transition is timely, or when a constraint is being honored in form but violated in substance. Ask: Is that a failure? Or is it an accurate description of the relationship between any documented process and the judgment required to execute it?

Closing (75–90 minutes)

Return to the framing question from the opening. What is this case about? Ask for three students to give a revised answer in light of class discussion. The goal is not consensus but precision. Each answer should be more analytically specific than the answers given at the opening.

Close with the question that should leave the room uncomfortable. If the most important knowledge in a complex production process is tacit and partially non-transferable, what does that imply for how we teach the design of such processes? Is this case an attempt to do something that cannot be done, or an illustration of how far documentation can go and where it must stop? Leave that question open.

Board Plan

The board should be organized in three columns, built progressively during class discussion. The following is a schematic; instructors should adapt to the flow of their specific section.

Column 1: Failure Modes and Their Pattern

- Compression: outputs technically responsive but analytically thin
- Tone drift: prose softens, hedging migrates, register shifts
- Rewriting when editing requested: local instructions, global interpretation
- Common pattern: instruction interpreted, not executed
- Implication: structural feature of the tools, not a communication failure

Column 2: Process Design Response

- Decomposition: outline before draft; single-purpose prompts
- Rooms: drafting / revision / evidence / reflection (functionally differentiated)
- Locking: fixed platforms; higher bar for reopening
- Provenance: tracks which system produced what, guides revision type
- Clean instance: independent reader; removes accumulated context
- Principles: role separation > tool specifics; functional differentiation > room specifics; bounded tasks; fixed foundations

Column 3: What Cannot Be Designed

- Tacit knowledge: the judgment embedded in execution
- Evaluation standards: what "good enough" means in context
- Timing judgment: when to transition, when to lock, when to reopen
- Psychological safety: willingness to see failures accurately
- Implication: description can convey structure; it cannot convey judgment

Instructor Guidance

On Managing the Opening

The opening question—what is this case about?—will typically produce thin answers on first pass. This is by design. Students who have prepared the case carefully will have their own framing; students who have read it more casually will reach for the surface description. Neither group should be corrected immediately. The value of holding multiple framings in parallel for the first ten minutes is that it creates a diagnostic for how the class will need to be moved. If most students are at the surface level, more time should be spent on Block 1; if most students are already at the process design level, Block 1 can be compressed and more time devoted to Block 3.

On the Prompt-Solution Fallacy

Expect this to surface strongly in Block 1, particularly from students with significant hands-on AI experience. The most effective intervention is not to refute the claim but to test it against the case evidence. What happened when Eccles added more explicit length specifications? What happened when he specified tone directly? Each answer in the case shows the same pattern—marginal improvement, problem not resolved—and accumulating those observations is more persuasive than abstract argument.

On the Tacit Knowledge Discussion

Block 3 has the highest risk of becoming abstract and losing analytical traction. The most effective anchor is the specific episode of the reflection room: Eccles uses the act of writing a prompt as a way of forcing precision about what he actually thinks is happening. Ask students to describe exactly what is happening in that room. The answer—that writing a description of a problem is a form of making tacit knowledge temporarily explicit, and that the value is partly in the writing rather than the response—is a concrete instantiation of Nonaka's externalization process and a bridge to the broader point about the limits of documentation.

On the HIT Framework

The concept of Hybrid Intelligent Teams (HITs), as developed in Eccles's SSRN working paper, provides a theoretical frame for the case's empirical evidence. Instructors who wish to use the paper should be selective. The relevant contribution is the concept of role-differentiated, human-orchestrated AI collaboration as a distinct organizational form, not a summary of the paper's full argument. The case is an instance of a HIT; the paper provides the vocabulary and the broader implications. Use the paper to help students see that what Eccles built is not idiosyncratic but generalizable—and that the design principles the case embodies are beginning to constitute a practical theory of knowledge production in human–AI systems.

On Self-Imposed Psychological Safety

The concept is typically introduced in organizational team contexts. Its application here to an individual practitioner's relationship with their own process and its failures is worth making explicit. Ask students: What would this workflow have looked like if Eccles had been unwilling to acknowledge that a section needed to be reopened, or that a room transition had come too late, or that the clean instance had revealed a genuine problem? The psychological safety frame illuminates not just the interpersonal dynamics of teams, but the epistemic disposition required for accurate self-evaluation in any complex production process.

On Classroom Composition

This case is appropriate for MBA students in courses on organizational design, knowledge management, operations, or technology strategy. It will also work effectively in executive education contexts where participants have direct experience managing AI-assisted workflows and can bring their own process failures into the discussion. In the latter setting, the study questions should be supplemented with a prompt asking participants to map their own workflows against the design principles Eccles developed.

Note on Production Method

This Teaching Note was produced using a Hybrid Intelligent Team operating under the same organizational principles the case describes. The production process is documented here both for transparency and because the failures encountered during production are themselves illustrative of the case's central arguments.

The HIT used to produce the case was designed for execution under constraint; the HIT used to produce the Teaching Note was designed for analysis, abstraction, and pedagogy. The difference lay not in the tools, but in the roles they were assigned and the process that structured their interaction. This distinction reflects a central implication of the Hybrid Intelligent Teams framework. HIT design is task-dependent, not tool-dependent.

The Team and Role Assignments

Five AI systems participated in distinct functional roles:

Gemini Pro — Analytical Foundation

Provided the initial analytical scaffold—identifying the case's core themes, key tensions, and the structure of the process design response—from a review of the case document.

Perplexity — Literature Grounding

Operated in a separate room, with no visibility into the Gemini output. Identified the relevant theoretical references—Polanyi, Nonaka, Simon, Baldwin and Clark, Galbraith, Brynjolfsson and McAfee, Edmondson—and confirmed their applicability to the case themes.

Claude — Synthesis

Received the analytical foundation and the literature grounding as separate inputs, alongside the case text and the structured specification for the Teaching Note. Produced the integrated document.

Dorothy (ChatGPT, memory-enabled) — Principal Collaborating Agent

Designed the prompt architecture for each stage of the process, structured the role differentiation across systems, and ensured that the Teaching Note remained consistent with the methodological logic developed in the case. Dorothy functioned as the system-level integrator, translating process design principles into executable prompt structures.

Clean ChatGPT (memory-disabled) — Independent Reader

Provided a context-free evaluation of outputs. By removing accumulated conversational context, the clean instance served as a diagnostic tool for identifying hidden assumptions, conceptual drift, or apparent clarity that depended on prior knowledge rather than the text itself. This role functioned as a control mechanism within the HIT.

The human user served as orchestrator throughout—assigning roles, structuring inputs, evaluating outputs, and making the integration and revision decisions that the specification required.

Observed Failures and Corrections

Two failures occurred during production. Both are reported here without mitigation.

The first was a task misinterpretation by Claude. On an initial pass, Claude produced a revised version of the case rather than a Teaching Note. The instruction had been clear; the output violated it. The failure was corrected by restructuring the input—separating the case document from the Teaching Note specification, adding an explicit constraint block to the prompt (“You are NOT editing or revising the case. You are writing a Teaching Note ONLY.”), and reissuing the task. The correction was complete on the subsequent pass. The episode is a direct instance of the case’s finding that AI systems interpret rather than execute instructions, and that the response to a structural misinterpretation is input restructuring rather than repeated restatement of the original instruction.

The second failure was a human omission. The user initially submitted the task without providing the case document as input, having assumed that Claude would work from the specification alone. The assumption was incorrect; the output was consequently thin and insufficiently grounded in the case evidence. The correction was to provide the case document as explicit input. The episode illustrates the case’s point about the indispensability of human orchestration: the most consequential single failure in this production process was not an AI misinterpretation but a human assumption about what the system would infer.

Connection to the HIT Framework

Both failures—and their corrections—are consistent with the principles described in Robert G. Eccles, *Hybrid Intelligent Teams* (SSRN Working Paper, 2026). The Eccles framework characterizes HITs as human-orchestrated systems in which the human’s primary contribution is not content generation but design—defining roles, structuring inputs, evaluating outputs, and maintaining the integrity of the production process across its stages.

The production of this Teaching Note provides a direct empirical instance of that characterization. Both failures originated in design decisions (task framing and input structuring), and both were corrected through redesign rather than through adjusting AI behavior. That the corrective mechanism was available—that the failures were diagnosable and the corrections were tractable—is itself evidence for the framework’s central claim. Human oversight in a HIT is not a supplement to AI capability but its enabling condition.