SDC Process: Applying Split-Domain Cognition to Any Domain

Canonical operational methodology. Version 1.0, April 2026.

Preface

About this document. This is the operational methodology — the process by which the SDC principle gets applied to a new domain. It is for practitioners who want to derive a domain-specific interpretation of Split-Domain Cognition for their own field, for Koher when partnering with such a domain, and for grant proposals or partnerships that name SDC as the underlying method. It pairs with the position essay (the *why*) and the long-form articulation (the *what*); this document is the *how*. The process is medium-agnostic — the AI pipeline is one application of it; the same process applies to a studio protocol, a curatorial framework, an editorial policy, an ethics audit, a public artwork, or a hiring panel.

Where this sits in the corpus. The canon home is splitdomaincognition.org. The position essay is *A Principle, Not a Pattern*; the long-form articulation is *Split-Domain Cognition*; the protocol for deriving a variant of SDC for a new domain is at /derivation-protocol-v1/; the catalogue of declared variants is at /variants/. A condensed entry into the principle is at /sdc-in-brief/.

Authority and version. Canonical, v1.0. April 2026. The website is the source of record. If this PDF and the website disagree, follow the website.

Use. Openly citable. See Governance for the one-person canon and how variants are admitted.

Author. Prayas Abhinav.

Introduction

A principle is portable; a process is what makes the portability real. SDC names a discipline — keep language work and judgement work distinct — but the discipline does not implement itself. The work of moving from "this is the principle" to "this is what the principle looks like in your domain" is concrete, and it has steps. The steps are not heavy. They are the ordinary moves a practitioner makes when adopting any structural commitment: locate where the collapse occurs in your domain, name the two domains in your domain's own vocabulary, design the holding-apart, build the artefact that carries it, test the artefact against the failure mode you started from. The pages that follow describe each step in the order in which a working group would meet them.

What this document is

`principle-not-pattern.md` argues that Split-Domain Cognition is a principle, not only an AI-pipeline pattern. This document specifies the *process* by which that principle gets applied to a new domain. It is for:

- Practitioners in any field (teaching, curation, editorial, ethics review, public art, hiring, research evaluation, governance) who want to apply SDC to their own work.
- Koher itself, when partnering with a domain to run the process together.
- Grant proposals and partnerships that claim SDC as the underlying method.

The process is medium-agnostic. The AI pipeline is one application of it. Everything below applies equally to a studio protocol, a curatorial framework, an editorial policy, an ethics audit, a public artwork, or a hiring panel.

Scope

When SDC applies

SDC is a useful process when **all three** of the following are true:

1. **Two kinds of cognition are in play.** The work involves both generative/interpretive/descriptive activity (language domain) *and* evaluative/verdictive/categorical activity (judgement domain). If only one is present, the process is not relevant.
2. **The two are at risk of collapsing.** In current practice in the domain, the two kinds of work are being done in the same call — by the same person, model, artefact, or moment — with no structural separation. Symptoms: verdicts that cannot be audited, descriptions that smuggle evaluation, rubrics that cannot be checked, rejections that cannot be learned from.
3. **The collapse produces a measurable cost.** Trust is eroding, decisions cannot be defended, students cannot learn the criteria, stakeholders cannot re-enter the argument, or the domain has drifted away from its own stated commitments.

If all three hold, SDC is an appropriate methodology. If only two hold, SDC may be overkill. If only one holds, SDC is not the right tool.

When SDC does not apply

SDC is not a universal framework. It is a narrow methodology for a specific kind of failure. It does **not** apply to:

- **Generative-only work.** Making a painting, writing a poem, improvising a performance, drafting a first sketch — these are language-domain activities that do not need a judgement layer at the moment of making. Applying SDC here would be a category mistake.
- **Judgement-only work with no language component.** A pure rule application with no interpretation (a tax calculation, a chess move, a database query) is already deterministic; there is nothing to separate.
- **Domains where the collapse is intended.** Some activities deliberately fuse language and judgement — political oratory, advertising, evangelism. SDC is a

refusal to do that fusion; applied where fusion is the point, it would dissolve the activity.

- **Decisions where speed matters more than auditability.** Emergency triage, split-second tactical calls, improvised responses. The process takes time to set up; if the time is not there, a heuristic is more appropriate.

Deliverables

A successful SDC application to a domain produces **five artefacts**. These are the concrete outputs of the process, regardless of medium.

1. Domain Map

A written description of:

- What the language-domain work in this domain consists of (what is being described, interpreted, generated).
- What the judgement-domain work in this domain consists of (what verdicts are being issued, against what criteria).
- Where the two are currently being collapsed, and what the cost of that collapse is.

Form: A document (typically 2–10 pages) that can be shown to any stakeholder. In AI tools, this becomes the architecture spec. In a design studio, it becomes a studio protocol. In a curatorial project, it becomes a curatorial brief.

2. Language Instrument

The artefact, process, or tool that produces structured descriptions from unstructured inputs. It does not judge; it describes.

Form varies by domain:

- AI pipeline: a Stage 1 classifier or qualifier (DeBERTa model, embedding function, LLM with structured output schema).
- Studio: a transcription + tagging protocol that captures what a student said or made without evaluating it.
- Curation: a description-first wall text convention that renders each work before positioning it.
- Editorial: a submission-summary template that states what a paper does before asking whether it should be published.
- Ethics review: an observation protocol that records what is happening before asking whether it is right.
- Hiring: an evidence-collection template that captures what a candidate demonstrated without yet ranking them.

3. Rubric (or Rule Set, or Policy)

The explicit judgement criteria. Authored in advance, held deterministically, applied uniformly across cases. This is the Stage 2 artefact.

Form varies:

- AI pipeline: `rules.py` with thresholds, priorities, severity levels.
- Studio: a faculty-authored rubric with specific criteria, weightings, and thresholds.
- Curation: a curatorial thesis with explicit inclusion/exclusion criteria.
- Editorial: editorial policy document with accept/revise/reject thresholds and reasons.
- Ethics review: a rights-based framework or harm-based framework, applied categorically.
- Hiring: a decision matrix with weighted criteria.

The rubric is the most consequential artefact in SDC. It is what most domains refuse to make explicit because explicitness creates accountability. SDC requires that accountability.

4. Narration Mechanism

The way the verdict gets communicated back to the person or situation it concerns. Stage 3. It does not re-judge; it renders the judgement legibly.

Form varies:

- AI pipeline: Haiku/LLM call at temperature=0 with the verdict and evidence as context.
- Studio: critique delivered after judgement has happened, referencing the rubric explicitly.
- Curation: exhibition text that names the curatorial thesis and explains each inclusion against it.
- Editorial: a decision letter that explains the verdict with reference to policy.
- Ethics review: a report that states findings with reference to the framework.
- Hiring: feedback to candidate, grounded in the decision matrix.

5. Audit Protocol

The procedure by which the whole pipeline can be inspected, challenged, and revised. It is what makes the domains' separation a durable commitment rather than a one-time design choice.

Form: A protocol that specifies (a) how the Language Instrument can be tested for drift or bias, (b) how the Rubric can be challenged and updated, (c) how the Narration can be checked for fidelity to the Rubric, (d) how stakeholders can raise concerns at each layer without collapsing them.

Without the audit protocol, the three stages will collapse over time under social pressure. The audit is the process's immune system.

Benefits

An SDC application, once deployed, confers **six benefits** on the domain. These are what the process is trying to produce and what justifies its overhead.

1. Auditability

Because the judgement layer is explicit and separate, any verdict can be opened and examined. This is the most immediate benefit. A decision that used to be "the expert said so" becomes "the rubric says X, the description was Y, so the verdict follows." The expertise is not removed — it is moved into the rubric, where it can be examined.

2. Reproducibility

Same inputs produce same verdicts. This is not a claim about mechanical determinism — the inputs themselves are produced by the Language Instrument, which is probabilistic or interpretive. But given the same described inputs, the judgement layer is deterministic. This makes disagreements locatable: is the disagreement about the description (Stage 1), the rubric (Stage 2), or the narration (Stage 3)?

3. Teachability

Explicit rubrics can be taught. Silent ones cannot. In a design studio, students who can read the rubric can learn what is being assessed; students who can only experience the critique cannot. In an editorial process, authors who see the policy can adjust to it; authors who only see rejection letters cannot. SDC makes the practice pedagogically accessible.

4. Refusal Capability

Because the judgement layer is explicit, the system can refuse to judge cases the rubric does not cover. It can say "this is outside the criteria" rather than producing a silent interpolation. This is a major benefit in ethics review, in curation, in editorial work — anywhere the pressure to render a verdict for everything produces worse decisions than acknowledged silence.

5. Trust Reconstruction

Domains that have lost stakeholder trust (through opaque decisions, appearing to contradict themselves, drifting from stated values) can recover trust by making the judgement layer visible. The trust gain is not from being more authoritative; it is from being more legible.

6. Refusal of Co-optation

When a rubric is explicit, it is harder to quietly rewrite under pressure. A domain that operates without an explicit rubric will drift under whoever's interests are loudest. A domain with an explicit rubric must negotiate any change publicly. SDC is therefore a form of structural protection for stated commitments.

The Process — Five Phases

The five phases move from diagnosis to commissioning. Each phase has a defined input, defined output, and a check before moving forward.

Phase 1 — Diagnosis (1–4 weeks)

Input: A domain, a felt sense that something is going wrong (verdicts cannot be audited; students cannot learn; authors cannot calibrate; stakeholders cannot re-enter the argument), a practitioner or team willing to look.

Work: Conversations with practitioners in the domain. What is the generative/descriptive work? What is the evaluative/verdictive work? Where are they collapsing? What is the cost? Often the collapse is invisible until it is named; the diagnosis phase's main deliverable is visibility.

Output: A one-page **Collapse Diagnosis**. States: the two domains in play, where they are currently collapsed, what the cost is, whether SDC is an appropriate methodology.

Check before Phase 2: Do the three fit conditions all hold (two kinds of cognition; collapse present; measurable cost)? If not, stop here.

Phase 2 — Separation Design (2–6 weeks)

Input: The Collapse Diagnosis.

Work: Design the three stages at the conceptual level. What should the Language Instrument do? What should the Rubric contain? What should the Narration Mechanism produce? At this stage nothing is built — only specified.

Output: A **Domain Map** (deliverable #1 above). Each stage is described in enough detail to build, but nothing is built yet. Stakeholders can review the design before construction begins.

Check before Phase 3: Does the design feel like it can be built? Does it feel like it will actually produce the benefits claimed? If not, iterate.

Phase 3 — Instrument Build (4–16 weeks)

Input: The Domain Map.

Work: Build the Language Instrument, author the Rubric, construct the Narration Mechanism. In AI tools, this is Stage 1 training, Stage 2 rules file, Stage 3 prompt. In a studio, this is the transcription protocol, the rubric document, and the critique template. In curation, this is the description convention, the curatorial thesis, and the wall-text template.

Output: The three working artefacts (deliverables #2, #3, #4 above). Each is usable in isolation and can be inspected.

Check before Phase 4: Can the three stages be exercised with real cases? Do they produce reasonable outputs?

Phase 4 — Interface and Audit (2–6 weeks)

Input: The three working artefacts.

Work: Define handoffs. How does Stage 1 output feed Stage 2? How does Stage 2 output feed Stage 3? What metadata travels between stages? Then design the audit protocol — how each stage can be inspected, challenged, revised.

Output: The **Audit Protocol** (deliverable #5). At this point the full process is documented and inspectable.

Check before Phase 5: Can a third party, reading the documentation, understand what the system does and why?

Phase 5 — Commissioning and Revision (ongoing)

Input: The full process.

Work: Put the process into live use. Monitor for drift. Revise the rubric when stated criteria change. Retrain or adjust the Language Instrument when drift appears. Audit periodically.

Output: A living practice in the domain, with the SDC artefacts as its substrate.

Check: Every 6–12 months, run the audit. If the collapse has returned, Phase 1 begins again.

Worked Examples

Each example shows what the five deliverables look like in one domain. The examples are intentionally brief; they are illustrations, not specifications.

AI Tool (Coherence Diagnostic)

Deliverable	Form
Domain Map	`architecture-spec.md` — three-layer pattern, canonical claim, failure modes

Language Instrument	DeBERTa classifier producing 5-dimensional structured scores
Rubric	`rules.py` with thresholds, severity levels, priority ordering
Narration Mechanism	Haiku call at temperature=0, streaming SSE, referencing scores and severity
Audit Protocol	Accuracy metrics on held-out test set; stage-by-stage inspectability in the UI; Behind the Curtain toggle

Design Studio (studioMeetingCcompanion)

Deliverable	Form
Domain Map	Faculty interview → transcript + problem description — "rubric is in my head; students cannot learn it"
Language Instrument	Studio meeting transcript + atom-extraction: what did the student say, what did they sketch, what decisions did they describe
Rubric	Faculty-authored rule set: for this studio, what counts as progress on each learning outcome, with thresholds
Narration Mechanism	Student-facing critique report, referencing specific atoms + specific rubric criteria
Audit Protocol	Faculty reviews the generated critique, can adjust the rubric, can flag atoms; student can contest the critique against the rubric

Curation (Museum-of-Vestigial-Desire mode)

Deliverable	Form
Domain Map	The curatorial question (what is this exhibition about?) + the refusal (what this exhibition will not do — e.g. not finish, not

	valorise, not resolve)
Language Instrument	For each work: a description convention that states what the work is in its own terms, before positioning
Rubric	The curatorial thesis as an inclusion/exclusion criterion — explicit, written, public
Narration Mechanism	Wall texts, catalogue, framing that references the thesis and each work's description
Audit Protocol	An external reader (advisor, critic) can check inclusions against the thesis; audience can raise works that seem to violate the criteria

Editorial Decision (Journal / Magazine)

Deliverable	Form
Domain Map	Current state: rejections feel arbitrary; authors cannot calibrate. Desired state: editorial policy is legible
Language Instrument	Submission summary template: what is this paper, what does it do, what does it claim
Rubric	Editorial policy: topic fit criteria, novelty thresholds, rigor thresholds, with accept/revise/reject bands
Narration Mechanism	Decision letter referencing policy and summary; explains the verdict against criteria
Audit Protocol	Editorial board periodically reviews a sample of decisions against policy; authors can appeal with reference to policy

Ethics Review (animalRightsLens, extendable to others)

Deliverable	Form

Domain Map	Current state: welfare language smuggled into rights territory; "humane meat" passes unchallenged. Desired state: rights and welfare distinguished
Language Instrument	Description of the practice or product — what is being done, to whom, under what conditions
Rubric	Rights-based framework: categorical criteria applied against descriptions
Narration Mechanism	Verdict with explicit reference to framework and description
Audit Protocol	Framework itself is publicly documented and revisable; verdicts can be challenged on description accuracy or framework application

Public Art (The Murmur Engine)

Deliverable	Form
Domain Map	Audience encounters a machine; machines usually render verdicts; this machine will refuse to
Language Instrument	The machine's sensors and model pipeline producing structured observations of its environment
Rubric	The machine's explicit refusal rules — when it will not output, when it will hesitate, when it will emit uncertainty rather than answer
Narration Mechanism	The gallery installation rendering the three layers visibly to the audience
Audit Protocol	The work is documented with its rules and instruments; audience can inspect the architecture; the artist can publish modifications

Hiring / Admissions

Deliverable	Form
Domain Map	Current state: hiring decisions rely on interviewer intuition; candidates cannot learn from rejection. Desired state: decisions are legible and teachable
Language Instrument	Evidence-collection template: what did the candidate demonstrate, in what form, against what prompts
Rubric	Decision matrix: weighted criteria, minimum thresholds per role, explicit accept/reject bands
Narration Mechanism	Feedback to candidate referencing evidence and matrix
Audit Protocol	Hiring committee reviews a sample of decisions against the matrix; candidates can request review; matrix itself is periodically revised

Research Evaluation (peer review)

Deliverable	Form
Domain Map	Current state: reviews are inconsistent; the same paper can receive contradictory judgements. Desired state: review criteria are explicit and applied uniformly
Language Instrument	Paper-description template capturing claims, methods, evidence, scope
Rubric	Review rubric: validity criteria, rigor criteria, novelty criteria, contribution thresholds
Narration Mechanism	Review letter referencing rubric and description

Audit Protocol	Editor can check reviews against rubric; authors can see rubric; reviewers themselves can be assessed

---

Engagement Model

A domain can engage with SDC at three levels. All three are legitimate; Koher supports all three.

Level A — Self-apply (free, open)

A practitioner reads this document, consults `principle-not-pattern.md`, and applies the process to their own domain on their own terms. Koher provides the writing, the worked examples, and the published tool case studies. No partnership required.

When this is right: The domain practitioner is already deeply familiar with their field, has authority to design the artefacts, and wants to keep the work inside their own institution. Most academic, editorial, and institutional applications will begin here.

Level B — Collaboration (funded by grants or partnerships)

A domain practitioner (individual or institution) partners with Koher to run the process together. Koher contributes the methodology, examples, and (where applicable) the software infrastructure. The domain partner contributes the domain expertise, the authority to author the rubric, and the institutional context.

Funding model: These collaborations are funded by grants targeting the domain (art grants for art applications, philosophy grants for philosophy work, ethics grants for ethics review). Koher does not charge for the methodology; the grant funds the collaboration. This preserves the "free tools, always" principle while making the work sustainable.

Examples already operative:

- Open Paws × Koher → *animalRightsLens* (ethics review domain; funded via SFF Thematic Round through Open Paws as fiscal sponsor)
- Shikha Parmar × Koher → *studioMeetingCompanion* (design studio pedagogy; funded via relevant education/pedagogy grants)
- FICA × Koher → *The Murmur Engine* (public art domain; funded via FICA Futures in Formation)

When this is right: The domain has a specific question, a practitioner willing to partner, and a grant context that fits.

Level C — Published Case (ongoing contribution to the practice)

Any Level A or Level B application can become a published case study, contributing back to the SDC practice and available to other practitioners. Case studies document: what the collapse was, what the process produced, what benefits emerged, what the limits were.

Form: A ~2,000-word case document, open-licensed, in the Koher GitHub organisation. Each case refines the process for future applications.

When this is right: Whenever the application has been running long enough for its benefits (and limits) to be visible — typically 6–12 months after Phase 5 commissioning.

What this process is not

- **Not a framework for all decision-making.** SDC is narrow. It addresses a specific kind of failure (language/judgement collapse). Decisions outside that failure mode do not need this process.
- **Not a replacement for domain expertise.** The Rubric is authored by domain experts. The process makes their expertise legible; it does not substitute for it. A bad rubric produces bad verdicts with perfect auditability — which is still bad verdicts.
- **Not an AI dependency.** None of the worked examples require AI except the AI-pipeline one. The process is medium-agnostic; software is one possible form for

the artefacts, not a requirement.

- **Not a neutrality claim.** Making the rubric explicit makes its commitments inspectable — which means contestable. SDC does not claim the rubric is neutral. It claims the rubric is legible enough to argue with.
- **Not a finished system.** This document is Version 1. Each application refines the methodology. Future versions will add what these examples omit.

Cross-references

- `principle-not-pattern.md` — the philosophical position this process operationalises.
- `sdc.md` — the long empirical articulation of the principle.
- `methodology-funding.md` — the Level 1 grounding paragraph and funder shortlist for grant applications using this process.
- `variants/` — domain-specific interpretations of SDC; each variant is a conceptual reading, this document is the operational methodology.
- `../positions-source/level-1-living-practice.md` — the paired position at the scale of the practice itself.
- `term-chosen.md` — the AI-pipeline technical definition.
- `../architecture-spec.md` — the AI-pipeline architecture (Koher's own Level B application of SDC to itself).
- `../processes/tool-development-process.md` — the Koher-internal process for AI-tool development.
- `../interests.md` — the range of domains where Prayas has genuine stake and SDC can legitimately be applied.

Version 1.0 — 15 April 2026. Relocated into `split-domain-cognition/` on 16 April 2026. To be revised as applications accumulate.