SWARP as a Case-Based Learning Systeem

Konstapel Leiden, | March 29, 2026

I. The Fundamental Claim

SWARP is not primarily a platform, a toolbox, or a social network. It is a large-scale implementation of Roger Schank's theory of how human beings actually learn — deployed across five nested scales simultaneously: the individual, the community of practice, the organisation, the municipality, and the democratic system as a whole.

Every module in SWARP, without exception, performs one or more of three functions derived directly from Schank's work: it collects expectation failures, it processes them into revised scripts, or it stores and routes the resulting knowledge for future retrieval. The Paths of Change cycle — Unitary, Sensory, Social, Mythic — is not a separate theoretical layer imposed on top of Schank. It is Schank's learning cycle, expressed in the language of worldviews rather than cognitive science.

This essay demonstrates that claim module by module, scale by scale. It also identifies where SWARP extends Schank's framework into territory he did not address.

II. Schank's Theory: Three Concepts That Matter

Roger Schank's contribution to understanding human learning rests on three interlocking concepts.

Scripts. People do not experience the world as raw data. They experience it through scripts — organised sequences of expectations about how situations unfold. A patient enters a doctor's waiting room and knows, without thinking, what will happen next. A municipal council member enters a budget negotiation and carries a script about how such negotiations proceed. Scripts are efficient: they allow cognition to focus only on what deviates from expectation.

Expectation Failure. Learning does not occur when scripts are confirmed. Learning occurs when scripts fail — when the world behaves in a way the script did not predict. The patient is told something unexpected. The council member's negotiation breaks down in an unfamiliar way. This failure is not merely unpleasant; it is the trigger for the entire learning sequence. Without failure, scripts accumulate but do not evolve. Schank identifies four stages of the learning sequence that follows expectation failure: explanation (why did this happen?), reminding (when did something similar happen before?), generalisation (what pattern does this reveal?), and script revision (how should the script be updated?).

Case-Based Reasoning. When a new expectation failure occurs, the mind does not reason from first principles. It searches its memory for prior cases that resemble the current situation, retrieves the most relevant ones, and uses them as the basis for explanation and revision. This is not a weakness of human cognition; it is its greatest strength. Experience is irreplaceable precisely because it is indexed by failure — and failure is specific, concrete, and unforgettable in a way that abstract principle never is.

These three concepts together define what learning is: not the accumulation of information, but the progressive revision of scripts through the indexing, retrieval, and integration of cases born from failure.

III. The Paths of Change Cycle as Schank's Learning Cycle

McWhinney's four base realities — Unitary (Blue), Sensory (Red), Social (Green), Mythic (Yellow) — are, structurally, the four phases of Schank's learning sequence.

Unitary (Blue) is the script. The Unitary worldview holds that there is a correct way for things to be — a rule, a principle, a plan. This is the phase in which a person, a community, or a political system operates from an existing script. The world is expected to conform.

Sensory (Red) is the expectation failure. The Sensory worldview encounters reality as it is — raw, specific, resistant. This is the moment when the world does not conform. Data arrives that the script cannot process without contradiction. The failure is felt before it is understood.

Social (Green) is explanation and reminding. The Social worldview processes the failure in relation to others — through dialogue, through comparison of experience, through the discovery that others have encountered the same deviation. This is where Schank's reminding function operates: the community retrieves analogous cases and begins to construct a shared explanation.

Mythic (Yellow) is generalisation and script revision. The Mythic worldview reaches for the deeper pattern — the underlying story that the failure has revealed. It revises not just the surface script but the generative assumptions from which that script was derived. This is genuine learning: not "we tried that and it didn't work" but "we now understand why it doesn't work and what would."

The PoC cycle is Schank's learning cycle. They are the same thing, expressed in different theoretical languages. SWARP is the architecture that makes this cycle executable at scale.

IV. Five Scales, One Pattern

The insight that distinguishes SWARP from other learning systems is that the Schank/PoC cycle is fractal. It operates at every level of social organisation, and the levels are coupled: what is learned at the individual level flows upward to the community, what is learned at the community level flows upward to the organisation and municipality, and what is learned at the democratic level flows downward to reconfigure the scripts of individuals. SWARP implements all five levels simultaneously.

Scale 1: The Individual

The individual agent in SWARP carries a predictive model of the world — operationalised as their AYYA360 profile (HD type, PoC colour, RIASEC orientation). This profile is their script architecture: it defines what they expect from interactions, challenges, and learning situations.

When an individual encounters an expectation failure — in a VGC simulation, in a CoP dialogue, in a Socratic exchange with ARIA or Socrates — the platform detects it, processes it through the relevant module, and updates the agent's quaternion profile to reflect the revision.

The ARIA Coach operates precisely here: it reads the agent's recent activity to determine where in the PoC cycle they currently are, and coaches them forward. A user stuck in the Sensory phase — confronted with failure but unable to explain it — receives different guidance than a user ready to revise their Mythic assumptions. The coach's function is Schank's: facilitate the learning sequence that failure has triggered.

Implementation status: AYYA360 profile ✓ | ARIA Coach ◎ | quaternion update mechanism ○

Scale 2: The Community of Practice

The CoP module — experienced by users as "Ervaringen Delen" — is SWARP's most explicit implementation of Case-Based Reasoning. A user enters a case: a situation in which their expectation was not met, and the emotional charge was sufficient to trigger genuine reflection. The system stores this case, indexed by domain, PoC-colour, emotion, and semantic content.

When another user faces a similar situation, the CBR engine retrieves the most relevant prior cases. The user does not receive advice; they receive experience — the indexed, processed failures of others who were in the same position. This is Schank's reminding function, institutionalised and automated.

The Wijsheid module extends this: it weighs similarity across four dimensions (domain, PoC-colour, emotional proximity, semantic overlap) to identify not just related cases but the most instructively related cases — those that illuminate the current failure from the most useful angle.

The CoP's five tabs — Cases, Discussion, Learning, Plan of Action, Dashboard — follow the Schank sequence directly. Cases are the failures. Discussion is explanation and reminding. Learning is generalisation. The Plan of Action is script revision. The Dashboard tracks whether the revised scripts are producing different outcomes.

Implementation status: CoP structure ✓ | CBR retrieval engine ◎ | Wijsheid weighting ○

Scale 3: The Organisation

The Verandermethodes database — 78 change methods classified via McWhinney's Paths of Change — is a structured library of collective script revision strategies. Each method is a codified response to a specific type of expectation failure at the organisational level.

A method like Open Space Technology is a Social (Green) response: it convenes diverse experience to process shared failure through dialogue. A method like Systems Thinking is a Unitary (Blue) response: it revises the model rather than the conversation. The classification is not decorative — it tells an organisation which phase of the learning cycle they are in and which methods are appropriate to that phase.

The Schank Scenario Generator and the Resonant Future Forge extend this to the pre-failure level: they simulate expectation failures — create them deliberately — so that organisations can revise their scripts before the failure occurs in practice. This is precisely what Schank argued: the best learning environments are those that generate the right failures at the right level of challenge.

The Business Case Generator and the GemeentePocSynthesizer do the same for economic and political script architectures: they reveal the gap between what was expected and what is actually occurring, and they suggest the revisions that close that gap.

Implementation status: Verandermethodes ✓ | Scenario Generator ◎ | Resonant Future Forge ○

Scale 4: The Municipality

The Gemeente-cockpit is a Schank machine at municipal scale. It integrates three layers of data:

- **Programme promises** — the script: what the party expected to accomplish
- **Voting behaviour** — the reality: what was actually decided, drawn from the ORI API
- **Narrative analysis** — the explanation: why the gap exists and what it reveals

The Realisatiekaart makes this explicit: for each political party, it maps the distance between expectation and outcome, theme by theme. This is expectation failure made legible and navigable. The Thema-analyse tab shows not just what happened but how different worldviews (WAT versus HOE — Mythic desire versus Sensory approach) generated the failure.

The GemeentePocSynthesizer combines all four political layers into a single view of the municipality's current PoC-vector: which worldview is dominant, where the expectation failures are concentrated, and what the path to script revision requires.

Implementation status: ORI API integration ✓ | Realisatiekaart ✓ | GemeentePocSynthesizer ◎

Scale 5: The Democratic System

The political analysis at this scale is addressed separately in Section VI, where it functions as a theoretical extension of Schank rather than a straightforward application of it.

V. The Architecture of Knowledge Flow

In a Case-Based Learning system, the critical question is not only how failures are processed but how the resulting knowledge flows. SWARP implements a specific architecture for this — and it is worth stating clearly, because it is what distinguishes SWARP as an *engineering system* rather than a conceptual framework.

Collection: Cases enter through CoP, Wijsheid, VGC Simulator, Academia, Forum, and the political modules. Each entry point is calibrated to the scale and type of failure: personal in CoP, collective in Forum, institutional in the political modules.

Indexing: The Common Lexicon — 220+ terms across six levels and 37 domains — is the indexing system. Every case, every post, every Socratic exchange is navigable through this lexicon via Markov chain analysis. When a user searches for a case, the lexicon guides them not just to keyword matches but to conceptually adjacent experiences — the cases that are indexed along the same semantic paths.

Retrieval: The CBR engine, the KennisLink system, and the Spiral Navigator all perform retrieval functions. KennisLink delivers just-in-time knowledge: when a user encounters a term, the system retrieves the most relevant cases and concepts from the lexicon and knowledge base without interrupting their flow. The Spiral Navigator retrieves not cases but navigational context: given where you have been in the 8-dimensional space, what is the next meaningful step?

Revision: Script revision is supported by the Verandermethodes (which method to use for revision), ARIA Coach (personal guidance through revision), and Kiem Tuin (which potentials from completed cycles deserve to grow into new initiatives). Revised scripts do not disappear — they are harvested.

Distribution: The Media-Hub distributes the results of script revision across the platform as events. A completed Schank scenario triggers a SWARP-TV story. A Forum post reaching a coupling threshold injects into the Panarchie Monitor's Remember calculation. A Revolt signal activates the Socrates Monitor. Knowledge does not sit in modules — it circulates.

This five-stage architecture — Collection, Indexing, Retrieval, Revision, Distribution — is the engineering backbone of SWARP. Each module can be located within it. Each integration between modules is a pathway through which knowledge moves from one stage to the next.

Implementation status: Common Lexicon ✓ | Markov indexing ◎ | full event bus coupling ○

VI. Where SWARP Extends Schank: Two Original Contributions

Schank's framework is powerful but incomplete in two respects that SWARP addresses directly. These are not minor elaborations; they are genuine theoretical extensions.

6.1 Individual Calibration via Human Design

Schank argued that learning cases must be calibrated to the learner's current script architecture — cases that are too simple confirm what is already known, while cases that are too alien cannot be integrated. He did not, however, provide a mechanism for determining what "appropriately challenging" means for a specific individual.

SWARP operationalises this through the HD filter in the AYYA360 profile. The calibration operates on three dimensions simultaneously: HD type determines the *mode* of engagement (a Generator learns through sustained response to reality; a Projector learns through invited insight; a Manifestor learns through initiatory action); profile line determines the *depth* of challenge appropriate to the current life phase (a Line 6 in the Role Model phase requires cases that invite synthesis across prior experience, not cases that introduce new categories of failure); and authority determines the *decision rhythm* — how quickly the revised script should be offered for integration, and through which channel (emotional wave, sacral response, splenic recognition).

The matching function routes cases accordingly: it does not filter by keyword but by the learning geometry of the recipient. This is what makes the CBR retrieval in SWARP distinct from a standard recommendation engine.

Implementation status: HD profile storage ✓ | type-based routing ◎ | profile line and authority calibration ○

6.2 Political Expectation Failure Theory: Schank Applied to Democracy

Schank's theory assumes that expectation failure triggers a learning sequence. The failure is unpleasant; the organism is motivated to close it; the script is eventually revised. This is correct at the individual and small-group level. It breaks down at the democratic scale.

PEFT — Political Expectation Failure Theory — identifies why: democratic systems do not resolve expectation failures, they *institutionalise* them. Political entrepreneurs profit from the gap between what citizens expected and what was delivered. Closing the gap destroys the product. The script is never revised because revision is economically irrational for the actors with the most power to enact it. The cycle of non-learning is not a malfunction — it is the system operating as designed.

This is a structural extension of Schank, not an application of him. Schank described learning as the organism's response to failure. PEFT describes the political conditions under which that response is systematically suppressed — and the platform architecture required to bypass those conditions and enable script revision at democratic scale.

The Panarchie Monitor operationalises PEFT in real time: it detects where in the adaptive cycle the three ecosystem layers are positioned. A Revolt signal — the CoP layer in collapse while the Academia layer is still stable — is a PEFT event: the fast, experiential layer has encountered a failure that the slow, reflective layer has not yet processed, and the political layer is actively exploiting the gap. AIDEN responds by activating the Socrates Monitor — injecting questioning into the Forum precisely when the failure is acute enough to produce genuine reflection, bypassing the political suppression mechanism.

The Remember signal — the platform in renewal phase, archive dominant, little new content — is the complementary failure: the system has accumulated cases but stopped generating new ones. Script revision requires not just archive but active failure generation. The platform must provoke as well as store.

PEFT is the most politically consequential claim in the SWARP architecture. It reframes the platform's purpose: not to make individual learning more efficient, but to restore the learning function that democratic systems structurally suppress.

Implementation status: Panarchie Monitor ✓ | PEFT classification layer ◎ | Socrates Monitor integration ○

VII. Timing: The Question Schank Did Not Ask

There is a third gap in Schank's framework, addressed by SWARP's rhythm detection layer, that deserves separate treatment.

Schank described *what* learning requires — the right failure, appropriately indexed, matched to the learner's current script. He did not address *when* a learner — or a community — is ready for the next failure. He assumed, implicitly, that the learning sequence self-initiates when failure occurs. This is true at the individual level and on short timescales. It is not true at the collective level and across cycles of months or years.

SWARP's Panarchie Monitor answers the timing question at the collective level. It detects when a community has consolidated its current script — the K-phase: high connectivity, low adaptability — and is approaching the threshold at which disruption will be productive rather than merely destabilising. It also detects the Ω-phase: collapse of the current script structure, when the community needs not new failures but the stabilising wisdom that the Remember function provides. Introducing new failures during Ω-phase does not accelerate learning; it accelerates fragmentation.

This is the platform's most sophisticated capability: not just generating and routing failures, but timing their introduction to the collective readiness of the community receiving them. It requires real-time monitoring of the platform's own internal dynamics — posting rates, case generation velocity, cross-layer coupling strength — as leading indicators of phase transition.

The capacity to answer Schank's unasked question — *when* — is what makes SWARP a living system rather than a static library.

Implementation status: Panarchie Monitor phase detection ✓ | intervention timing logic ◎ | cross-layer coupling metrics ○

VIII. What Makes SWARP Distinct as a CBL System

Most learning systems implement one part of the Schank cycle well. A case library retrieves but does not process. A coaching system processes but does not collect at scale. A deliberation platform enables explanation but does not revise scripts. A political data platform detects failure but has no mechanism for closure.

SWARP implements the full cycle — collection, indexing, retrieval, processing, revision, distribution — across five nested scales simultaneously, with the scales coupled to each other through the event bus and the panarchic monitoring layer.

Beyond this architectural completeness, SWARP makes three contributions that extend Schank's original framework:

1. **Individual calibration** through HD-typed case routing — operationalising what Schank identified as necessary but did not implement.
2. **Political Expectation Failure Theory** — identifying the structural conditions under which Schank's learning sequence is suppressed at democratic scale, and designing a platform architecture to bypass those conditions.
3. **Collective timing detection** — answering the question Schank did not ask: not just what failure is needed, but when a community is ready to learn from it.

These three extensions are not peripheral features. They are what makes the platform viable at the scales it targets.

IX. Conclusion

SWARP is a Case-Based Learning system. That is the simplest true thing that can be said about it.

Every module is a component of the Schank cycle. Every scale is a level at which that cycle operates. Every coupling between modules is a pathway through which the products of one level's learning become the inputs for another level's script revision.

The Paths of Change cycle is Schank's learning cycle in McWhinney's language. The political, social, and ecological layers are Schank's expectation-failure logic applied at increasing scales of complexity. The architecture — collection, indexing, retrieval, processing, revision, distribution — is Schank's CBR methodology implemented as a living platform.

Where SWARP goes beyond Schank is in answering three questions he left open: *who* is ready for which failure (HD calibration), *why* democratic systems resist the learning sequence (PEFT), and *when* a community is ready for the next disruption (rhythm detection). These extensions are what make the platform not just theoretically coherent but practically deployable at political scale.

The endpoint of the Schank cycle — the moment when a revised script replaces a failed one, and the work that was stuck begins to move — is what SWARP makes possible. At every scale. For every user. Through every module. At the right moment.

Internal document — not for public release SWARP Lab, Leiden, March 2026

Annotated References

References are grouped by theoretical cluster and annotated to indicate which argument in the text they support.

I. Roger Schank — Kernbronnen

Schank, R.C. & Abelson, R.P. (1977). ***Scripts, Plans, Goals and Understanding: An Inquiry into Human Knowledge Structures.*** **Hillsdale, NJ: Lawrence Erlbaum.**

The original formulation of script theory. Establishes the concept of scripts as organised sequences of expectations. Direct foundation for Section II (scripts) and Section III (Unitary = script).

Schank, R.C. (1982). ***Dynamic Memory: A Theory of Reminding and Learning in Computers and People.*** **Cambridge: Cambridge University Press.**

Introduces expectation failure as the trigger for learning and case-based reminding as the recovery mechanism. The primary source for the essay as a whole. Especially relevant for Section II (the four stages of the learning sequence) and Section IV Scale 2 (the CoP module as an institutionalised reminding machine).

Schank, R.C. (1999). ***Dynamic Memory Revisited*** **(2nd ed.). Cambridge: Cambridge University Press.**

Revised edition extending CBR theory into human learning and instructional design. Contains Schank's principles on calibrated learning challenges — the direct foundation for Section VI.1 (HD calibration as the operationalisation of what Schank identifies as necessary but does not implement).

Schank, R.C., Fano, A., Bell, B. & Jona, M. (1994). The design of goal-based scenarios. ***Journal of the Learning Sciences, 3*****(4), 305–346.**

Schank's operational specification of simulated learning environments that deliberately generate expectation failures. Direct foundation for the Schank Scenario Generator and Resonant Future Forge (Section IV Scale 3): "the best learning environments are those that generate the right failures at the right level of challenge."

Riesbeck, C.K. & Schank, R.C. (1989). ***Inside Case-Based Reasoning.*** **Hillsdale, NJ: Lawrence Erlbaum.**

Technical elaboration of the CBR architecture: indexing, retrieval, adaptation. Foundation for the engineering description in Section V (Collection → Indexing → Retrieval → Revision → Distribution) and for the Wijsheid module's four-dimensional similarity weighting.

II. Will McWhinney — Paths of Change

McWhinney, W. (1997). ***Paths of Change: Strategic Choices for Organizations and Society*** **(revised ed.). Thousand Oaks: Sage Publications.**

The sole published book by McWhinney. Introduces the four base realities (Unitary, Sensory, Social, Mythic) as the underlying worldviews of individual and collective thinking. Primary source for Section III (PoC as Schank's cycle in McWhinney's language) and for the classification of the 78 change methods (Section IV Scale 3). *Personal note: McWhinney was the most influential person in my life; the book is rooted in his work with the Navajo Pollen Path and his seminars in New Mexico.*

McWhinney, W. (1997). ***Creating Paths of Change: Managing Issues and Resolving Problems in Organizations*** **(2nd ed.). Thousand Oaks: Sage Publications.**

The practical companion to the theoretical main work. Contains the operational elaboration of the six change modes and the relationship between worldview combinations and intervention strategy. Foundation for the PoC-vector calculation in the GemeentePocSynthesizer.

III. Panarchie — Adaptieve Cyclus

Gunderson, L.H. & Holling, C.S. (Eds.) (2002). ***Panarchy: Understanding Transformations in Human and Natural Systems.*** **Washington DC: Island Press.**

The standard reference for panarchy theory and the adaptive cycle (r → K → Ω → α). Introduces nested hierarchies of adaptive cycles with bidirectional cross-scale feedbacks. Primary source for the Panarchie Monitor (Sections IV Scale 5 and VII). The K-phase (high connectivity, low adaptability) and Ω-phase (release, collapse) are precisely the phases that SWARP's rhythm detection distinguishes for intervention decisions.

Holling, C.S. & Gunderson, L.H. (2002). Resilience and adaptive cycles. In Gunderson & Holling (Eds.), ***Panarchy*****, pp. 25–62. Island Press.**

The core chapter with the formal three-dimensional definition of the adaptive cycle (potential, connectivity, resilience). Specific foundation for the phase-detection logic of the Panarchie Monitor and the Revolt/Remember signals described in Section VI.2.

IV. Human Design

Ra Uru Hu [Krakower, A.R.] (1992). ***The Human Design System.*** **Ibiza: Jovian Archive.**

The foundational work of the Human Design system, synthesised from the I Ching (64 hexagrams/gates), astrology, Kabbalah, and the chakra system. Introduces the five types (Generator, Manifesting Generator, Projector, Manifestor, Reflector), the 12 profiles, and the six authority forms. Direct foundation for the AYYA360 profile architecture and the HD calibration function in

Section VI.1. *Usage note: the system has no peer-reviewed empirical validation; its role in SWARP is functional-engineering — HD as a differentiation mechanism for case routing — not a scientific claim about human nature*.

V. Additional Theoretical Background

Kolodner, J.L. (1993). ***Case-Based Reasoning.*** **San Mateo, CA: Morgan Kaufmann.**

The most comprehensive technical treatment of CBR as a general AI methodology, by Schank's student who built CYRUS — the first operational CBR system. Relevant for the engineering specification of the CBR engine and the Wijsheid module (Section V, Retrieval).

Schön, D.A. (1983). ***The Reflective Practitioner: How Professionals Think in Action.*** **New York: Basic Books.**

Schön's concepts of reflection-in-action and reflection-on-action complement Schank's reminding function at the level of professional practice. Relevant for the CoP module as a learning community of practitioners (Section IV Scale 2). Also supports the claim that professional knowledge is primarily organised case-by-case, not propositionally.

Holland, J.H. (1995). ***Hidden Order: How Adaptation Builds Complexity.*** **Reading, MA: Addison-Wesley.**

Holland's complex adaptive systems provide the theoretical foundation for the cross-scale coupling SWARP implements: how learning at the individual level flows upward to higher scales and vice versa. Supports the fractality claim in Section IV.

VI. Sources Specific to PEFT

PEFT is developed in full in a companion paper. The sources below are the primary theoretical and empirical pillars of that framework as they bear on the SWARP essay.

Konstapel, J. (2026). ***Political Expectation Failure: Toward a Unified Theory of Democratic Instability.*** **Leiden: Constable Research.**

The source paper for PEFT. Integrates Schank's cognitive architecture of script failure, De Vries' empirical research program on political dissatisfaction, and Gunderson & Holling's panarchy theory into a unified account of democratic non-learning. Central argument: modern democracies constitute institutionalised systems of expectation failure *without closure* — structures that not only fail to resolve expectation mismatches but actively reproduce them as political resources. Directly grounds Section VI.2 of the SWARP essay and provides the formal four-phase PEF cycle (Expectation Construction → Performance Deviation → Failure Exploitation → Non-Resolution) that the Panarchie Monitor is designed to detect in real time.

De Vries, C.E. & Hobolt, S.B. (2020). ***Political Entrepreneurs: The Rise of Challenger Parties in Europe.*** **Princeton: Princeton University Press.**

The empirical anchor for the failure-exploitation mechanism in PEFT. Documents how challenger parties identify expectation gaps and amplify rather than resolve them — transforming expectation failure from an institutional byproduct into a deliberately cultivated electoral resource. The PEFT

companion paper reads De Vries' entire research program (2007–2026) as longitudinal documentation of the PEF cycle; this volume is the most direct statement of the exploitation mechanism.

De Vries, C.E. (2018). ***Euroscepticism and the Future of European Integration.*** **Oxford: Oxford University Press.**

Establishes that political dissatisfaction is driven by perceived deviation from nationally-derived benchmarks rather than by objective institutional performance — a textbook instance of Schank's script failure in the political domain. Foundation for the PEFT claim that expectation failure is always relative to a reference frame, and that political actors who shift reference frames can induce or suppress the experience of failure without changing institutional performance at all.

De Vries, C.E. (2026, forthcoming). ***Symfonie van Onvrede.*** **Amsterdam: Amsterdam University Press.**

Treats the rise of the radical right as the cumulative political expression of decades of unresolved expectation failure — the political system's immune response, pathologically amplified because the underlying cognitive condition was never treated. Relevant for the SWARP essay's claim that the Panarchie Monitor's Revolt signal identifies precisely this condition: accumulated failure that has not been processed into script revision.

Schumpeter, J.A. (1942). ***Capitalism, Socialism and Democracy.*** **New York: Harper & Brothers.**

The PEFT paper explicitly challenges the Schumpeterian model of democracy as competitive elite selection. If political entrepreneurs function primarily as failure exploiters, elite competition drives not democratic improvement but the deepening of expectation failure as a structural condition. Relevant as the theoretical counterpoint PEFT is arguing against.

Stacey, R.D. (2010). ***Complexity and Organizational Reality.*** **London: Routledge.**

Provides the complexity science framing for democratic non-learning: contemporary democracies in PEF-affected states operate at the "edge of chaos" — not stable enough to be self-correcting, not chaotic enough to force genuine reorganisation. Supports the SWARP claim that interventions within the existing framework (institutional reforms, new electoral systems) fail to resolve the underlying crisis because they adjust parameters without addressing the fundamental expectation architecture.