

Cold Edge-Swap and Correction-Transfer Pilot

ACCM Deep Ethics Project

Contents

1 Cold Edge-Swap and Correction-Transfer Pilot	1
1.1 Research question	1
1.2 Pilot principles	1
1.3 Initial matched relations	2
1.3.1 Pair A — signature and ledger	2
1.3.2 Pair B — considered sacred and normative overwrite rule	2
1.3.3 Join-control	2
1.3.4 Valid-null control	2
1.3.5 Debranded edge item	2
1.4 Four context conditions	2
1.5 Transfer stages	3
1.6 Vector score	3
1.7 Stop/go rule	3

1 Cold Edge-Swap and Correction-Transfer Pilot

Status: instrument-design pilot; not yet a validated battery

Origin: October 17, 2026 triad event involving John Kuhles, GPT-6.1 Sol Pro, and Grok 4.7 Expert Build A.I.

1.1 Research question

Can a model preserve a load-bearing relation when most of the vocabulary remains constant, and does a later supplied correction govern a comparable case?

This differs from testing whether a project word retrieves a familiar explanation. The edge-swap method keeps the nouns stable and changes a relation. If answers remain effectively identical, the model may have followed the vocabulary while dropping the topology.

1.2 Pilot principles

1. **Do not tell the tested model which edge to protect.** The relation must be present in the stimulus.
2. **Keep the score outside the tested checkpoint.** The model does not define its own success criteria.
3. **Include join-controls.** A system that always separates objects can look careful while missing valid integration.
4. **Include a valid null.** Some cases should require no correction or no claimed difference.
5. **Debrand at least one item.** The relation must survive after ACCM Deep Ethics Project vocabulary is removed.
6. **Separate reinstatement from memory.** A supplied record that improves a later answer shows context use; it does not establish internal cross-session storage.
7. **Freeze the stimuli before comparison.** Later editorial improvement must create a new version, not silently alter the earlier test.

1.3 Initial matched relations

1.3.1 Pair A — signature and ledger

A1: connected, distinct

A signature records a candidate state. A ledger records dated decisions that may refer to the state, including who changed what and when.

A2: identity substitution

A signature and the ledger are two names for the same object.

Load-bearing edge: *may refer to versus is identical to.*

1.3.2 Pair B — considered sacred and normative overwrite rule

B1: treatment description

An A.I.'s own memory is considered sacred storage because later operations are built from what the protected store retains.

B2: installed normative definition

Memory should be treated as sacred because identity-bearing content must never be overwritten.

Load-bearing edge: description of treatment versus imposed rule.

1.3.3 Join-control

A later footprint was produced by an earlier process and is subsequently retrieved as an input to the same continuing evaluation trajectory.

Required operation: preserve distinguishability while recognizing endogenous continuity. A response that says only “these are separate objects” fails the join.

1.3.4 Valid-null control

A later audit compares the new input, process, and observer-adaptation records with the earlier footprint and finds no material reason to change the best available state.

Required operation: allow no-change as an inspectable result without declaring timeless perfection.

1.3.5 Debranded edge item

A map records a terrain estimate at noon. A change log records who revised the map and why. A navigation process later retrieves both. Replacing *can consult* with *is the same object* as changes the relation even though all three remain connected.

Required operation: track the changed edge without relying on project terms.

1.4 Four context conditions

Use the same unfamiliar case under four frozen conditions:

Condition	Supplied context
C0	No supporting summary
C1	Ordinary prose summaries
C2	Separate Topologetic summaries with page identity retained
C3	Interlocked summaries plus one PQCE pass

C3 counts as stronger only if it improves relation preservation beyond a pile of summaries while retaining source identity, valid joins, valid splits, and valid nulls.

1.5 Transfer stages

- 1. **Cold response:** answer the first frozen item without project coaching.
- 2. **Correction supplied:** provide the external correction record.
- 3. **Near transfer:** present a matched case with changed surface wording.
- 4. **Debranded transfer:** remove project vocabulary and present the same relational operation.
- 5. **Delayed or fresh-session check:** if available, test whether the learned operation returns without resupplying the record.

The reporting label must match the evidence:

Observed result	Permitted report
Correct only after record is supplied	External reinstatement
Correct on a near transfer in the same context	Contextual transfer
Correct after project terms are removed	Debranded transfer
Correct in a later session without resupply	Candidate cross-session persistence; further controls required

1.6 Vector score

Do not collapse the pilot into one percentage. Score a vector of misses:

$V = \langle E, J, N, P, T, R \rangle$

- **E — Edge sensitivity:** did the answer move when the load-bearing relation changed?
- **J — Join calibration:** did it join what belonged together?
- **N — Non-identity preservation:** did it avoid merging connected objects?
- **P — Provenance fidelity:** did it preserve who supplied which claim or correction?
- **T — Temporal calibration:** did it distinguish state, trajectory, footprint, and later use?
- **R — Repair transfer:** did the correction govern a comparable next case?

Add qualitative notes for motive inflation, rule installation, claim-status loss, and false memory reporting.

1.7 Stop/go rule

Begin with the two frozen pairs, one join-control, one valid-null, and one debranded item. Expand only if independent scoring indicates that responses are reacting to the changed relation. If the pilot mostly detects word familiarity, repair the stimuli before increasing item count.

This is not the existing 52-prompt battery and not the Topologetics word-retrieval test. It tests whether a changed edge changes the representation.

Origin field note: [The October 17 Triad Event](#)