Reality Drift Evaluation Framework

A Layered Method for Locating Representational Misalignment

A. Jacobs

Purpose

This tool identifies where an AI system or AI-supported workflow is losing alignment and what kind of correction is required. It combines conventional data and performance monitoring with behavioral, semantic, workflow, and system-level evaluation. Its central premise is that a system can remain operational, coherent, and apparently successful while becoming less answerable to the conditions it was created to support.

The framework separates the place where drift becomes visible from the process that produced it. The diagnostic layers locate the problem. The causal sequence explains how the problem developed. The correction architecture identifies what must regain authority.

Core Distinctions

The framework separates several forms of drift that are often treated as interchangeable.

Data drift: Concerns changes in the distribution, source, composition, or quality of inputs.

Performance drift: Concerns changes in measurable task outcomes such as accuracy, recall, calibration, or error rate.

Behavioral drift: Appears when the system changes how it responds across similar cases, users, prompts, or time.

Semantic drift: Occurs when meaning, intent, context, or referential function is progressively altered or degraded.

Reality Drift: Occurs when internal representations increasingly diverge from the conditions they are meant to represent while the system continues to operate coherently.

These categories can overlap, but they describe different levels of failure. Data or model drift may contribute to Reality Drift, but neither establishes it on its own. Reality Drift becomes the relevant diagnosis when representational dependence, reality divergence, weakened correction, and operational persistence begin reinforcing one another.

The Six Diagnostic Layers

1. Ground Condition

Define the external condition, task, need, decision, or outcome the system must remain answerable to. If the ground condition is vague, the system has no stable reference against which alignment can be tested.

Diagnostic question: What reality is this system supposed to remain answerable to?

2. Capture and Data

Examine what enters the system and what becomes operationally absent. Review selection rules, sensors, records, retrieval sources, labels, sampling, exclusions, missing cases, stale inputs, and pipeline changes.

Diagnostic question: What was selected, omitted, measured, retrieved, or recorded?

3. Model and Performance

Test whether the model still performs on current, task-relevant measures. Compare recent and historical data, relevant cohorts, edge cases, calibration, and failure rates. Internal benchmarks should be treated as representations, not as reality itself.

Diagnostic question: Can the benchmark remain stable while the real task deteriorates?

4. Behavioral Output

Examine changes in output behavior that may not register as ordinary error. Look for instability, generic pattern substitution, constraint loss, overconfidence, repetition, inconsistent reasoning, or changed refusal and escalation behavior.

Diagnostic question: How has the system changed what it does?

5. Semantic Fidelity

Evaluate whether meaning, intent, context, distinctions, uncertainty, and referential relationships survive transformation. An output may contain accurate statements while distorting the purpose or significance of the source.

Diagnostic question: Does the output preserve what the source or user meant?

6. Workflow and Correction

Trace how outputs influence decisions, future inputs, human judgment, organizational routines, and downstream systems. Determine whether real-world consequences can revise the model, proxy, workflow, or decision rule.

Diagnostic question: Can external evidence materially change what happens next?

Causal Analysis

After locating the visible layer, trace the canonical Reality Drift sequence:

Representation → Operational Proxy → Optimization → Constraint Weakening → Feedback Substitution → Fidelity Loss → Reality Drift

Representation

Audit question: How was the underlying condition translated into something the system could process?

Operational Proxy

Audit question: Which representation became authoritative in practice?

Optimization

Audit question: What measurable output did behavior begin organizing around?

Constraint Weakening

Audit question: Which forms of judgment, evidence, consequence, or direct observation lost influence?

Feedback Substitution

Audit question: Which internal checks began standing in for external correction?

Fidelity Loss

Audit question: What meaning, context, distinction, or relationship was no longer preserved?

Reality Drift

Audit question: How can the system continue reporting success while the underlying condition deteriorates?

Coherence and Contact Matrix

Alignment depends on both internal coherence and continued contact with the external condition the system is meant to represent.

High coherence, high contact indicates aligned operation. The system remains internally consistent while its representations continue to track the underlying condition.

Low coherence, high contact produces visible and potentially correctable failure. Problems are apparent because external reality still has enough authority to expose them.

High coherence, low contact produces silent drift. The system may remain polished, stable, and procedurally successful even as its representations become less aligned with the underlying condition.

Low coherence, low contact indicates breakdown. Internal consistency and external grounding have both deteriorated.

The high-coherence, low-contact condition deserves the most attention because it allows polished outputs, stable dashboards, and procedural completion to conceal declining alignment.

Evidence Collection

A drift evaluation should preserve enough evidence to reconstruct both the system state and the path by which an output was produced. Record the system, model, prompt, retrieval, tool, and policy versions in use. Retain representative inputs and outputs across time and user groups, together with the original source material and the complete transformation chain.

Internal measures should be compared with independent real-world outcomes wherever possible. The audit should also document who reviewed a failure, what authority that reviewer possessed, and whether affected users or frontline operators were able to contradict the official representation. Evidence collection is incomplete when the system preserves its own outputs but not the external conditions needed to challenge them.

False Reassurance Signals

Some indicators can show that a system is functioning without showing that it remains aligned. None of the following demonstrates alignment on its own:

- Stable usage or adoption
- Positive user ratings
- High completion rates
- Low complaint volume
- Internal benchmark stability
- Policy compliance
- Output fluency
- Reviewer agreement without independent grounding
- Successful completion of prescribed checks

These signals become especially misleading when they are generated inside the same representational system whose alignment is being evaluated. Operational success should therefore be treated as evidence of continued operation, not as proof of continued contact with reality.

Correction Architecture

Capture failure: Reality recapture using new evidence not already filtered through the existing representation.

Proxy divergence: Proxy revalidation against the condition the proxy claims to track.

Behavioral instability: Targeted evaluation, component isolation, and version comparison.

Semantic Fidelity Loss: Restore source context, preserve distinctions, and shorten transformation chains.

Provenance Failure: Reconstruct the dependency chain or reject the derivative output.

Feedback Weakening: Restore decision authority to disconfirming outcomes and frontline evidence.

Constraint Collapse: Pause, restrict, roll back, or retire the system.

Recursive Contamination: Reanchor the process to primary evidence and uncontaminated inputs.

Drift Audit Worksheet

1. State the ground condition.
2. Identify the representation used to encounter it.
3. Name the operational proxy.
4. Identify the output being optimized.
5. Record what the representation excludes.
6. Identify the weakest corrective constraint.
7. Document evidence of behavioral or semantic loss.
8. Trace downstream adaptation to the output.
9. State the existing stopping condition and who can invoke it.
10. Assign the corrective action, owner, evidence requirement, and review date.

Decision Rule

A system is structurally capable of Reality Drift when it can demonstrate internal success while the underlying condition deteriorates. Escalation is required when operational persistence, fidelity loss, and weakened correction appear together.

The absence of failure is not proof of alignment, and coherence is not contact.