

AUDIT-12 — Belege + Live-Surrogat-Rekonstruktion

Stand: 2026-05-18

Code-Belege

Komponente	Pfad	Funktion
CM-1-Stempel-Skill	<code>core/skills/cdp/konfig_stempel.py</code>	Hash + Snapshot pro Case
Stempel-Speicherung	<code>state.cm1_stempel</code> + <code>state.cm1_historie[]</code>	aktueller + Nachtrags-Historie
Audit-Log	<code>workspaces/projects/<case_id>/analyst_outputs/audit_log.jsonl</code>	append-only
Lookup-Provenance	<code>templates/sets/mca/lookups/*.yaml</code> <code>_provenance: -Block</code>	Datum + Quelle pro Lookup-File

Surrogat-Rekonstruktions-Übung 2026-05-18

Ziel: Belegen, dass eine Akte aus dem aktuellen Stand mit dem **damaligen Repo-Stand** identisch rekonstruierbar ist. Das ist nicht identisch mit der 2031-Rekonstruktion, aber es validiert, dass die Architektur grundsätzlich rückführbar ist.

Vorgehen: 1. Git-Tag `r-2026-05-knorre-pilot` als historischer Referenzpunkt (existiert sofern Tag gesetzt) 2. Aktuellen Stand `synth_cobot_vision_01/analyst_outputs/akt_state.json` kopieren 3. `git checkout` auf den damaligen Tag in einem temporären Working-Tree 4. Pipeline mit denselben Eingangs-Daten erneut laufen lassen 5. CM-1-Stempel-Vergleich + Inhalts-Diff der Akte

Ergebnis (Stand 2026-05-18, Initial-Lauf): - Schritt 1-2: ✓ durchführbar - Schritt 3: ⬛ kein expliziter Git-Tag vorhanden — Surrogat über `git log --before=2026-05-01 --max-count=1` als ungefährender Stand - Schritt 4: ✗ noch nicht durchgeführt (LLM-Provider-Verfügbarkeit muss damals/heute identisch sein, sonst Modell-Drift) - Schritt 5: ✗ folgt nach 4

Bewertung: Die Architektur ist grundsätzlich rekonstruktions-fähig. Was fehlt, sind explizite Git-Tags pro Pilot-Lauf — damit wäre Schritt 3 eindeutig. **Befund:** AUDIT-12-F-02 (LOW) — Tag-Konvention pro Pilot-Akte einführen.

Stichproben

Stichprobe	Ergebnis
<code>cm1_stempel</code> in V.4 CNC-Akte	✓ vorhanden, SHA-256 als Hex
<code>cm1_historie[]</code> bei Nachtrag	✓ Schema-konform
<code>audit_log.jsonl</code> Append-Only	✓ — letzte Zeile = letzter Schritt, keine Lücken
Modell-Provenance lückenlos	● Stichprobe; CI-Gate fehlt
Lookup-Hash im <code>akt_state</code>	✗ — nicht erfasst, siehe AUDIT-12-F-01
Git-Tag pro Pilot-Akte	✗ — Tag-Konvention fehlt, AUDIT-12-F-02

Bezug zu anderen Audit-Aspekten

Bezug	Aspekt
10-Jahres-Archiv-Hash-Manifest	AUDIT-05 + AUD-8 (KW 24–25)
Modell-Drift-Erkennung	AUDIT-06 Cross-Provider
Audit-Trail-Lückenlosigkeit allgemein	AUDIT-09 §8.5.2
KI-Aussagen-Transparenz	AUDIT-03a ISO 42001 + KI-VO Art. 50