Lineage Map Specification

Purpose

A lineage map is the directed graph from a row in the Source Inventory to a model output a user, regulator, or paying customer actually sees. If a builder cannot draw the map for a given output, that output is unsupported, and the system must treat it as such per s3-decision-traceability.

This spec defines the required node types, edge labels, and conventions so every GROW module produces lineage that is mechanically comparable across runs and across builders.

Required Node Types

Every lineage map MUST use these node types and only these node types. Subtypes are allowed; new top-level types are not.

Node type	Symbol	Definition
Source	rectangle	A row from the Source Inventory. Labeled with source_id.
Extraction	parallelogram	Raw pull from a source (API call, file read, query). Records timestamp, version of the source.
Transformation	hexagon	Any deterministic or near-deterministic processing step: join, filter, normalization, deduplication, enrichment.
Embedding	trapezoid	Vectorization step. Records embedding model + version.
Retrieval	diamond	Selection of a subset of records to surface to a model or rule. Records retriever + parameters.
Inference	rounded-rect	LLM or model call. Records model id, version, temperature, prompt hash.
ToolCall	cylinder	External tool invocation (search, calculator, MCP tool, code exec).
Decision	double-rect	The point at which decision_origin is recorded per the C2 enum.
Output	circle	The artifact delivered downstream. Hashed and stored.

Required Edge Labels

Edges are not decorative. Every edge in a GROW lineage map MUST carry at least:

- operation — what the edge does (e.g., filter:date>=2026-01-01, join:on=parcel_id, summarize:tokens<=400).
- data_shape — what passes through (e.g., rows:1284, tokens:9300, vectors:512x1).
- confidence_propagation — how the edge moves confidence_band (preserve | downgrade | upgrade-on-merge). Upgrades require an explicit merge rule.
- is_deterministic — true | false. Inference and many retrievals are false; this drives s3-reproducibility-controls.

Worked End-to-End Lineage

The diagram below traces one operational output: an automated weekly Stormwater Grant Status Memo drafted for the District Commission and reconciled against parcel impacts. It crosses three sources (fl-dep-lpa0381, wcpa-parcels, ecoguardian-stream).

```
flowchart TD
    S1[Source: fl-dep-lpa0381]:::source
    S2[Source: wcpa-parcels]:::source
    S3[Source: ecoguardian-stream]:::source

    E1[/Extraction: DEP ledger pull 2026-05-25/]:::extract
    E2[/Extraction: parcel snapshot 2026-05-25/]:::extract
    E3[/Extraction: 7-day sensor window/]:::extract

    T1{{Transformation: normalize disbursement rows}}:::xform
    T2{{Transformation: join parcels x impact zones on parcel_id}}:::xform
    T3{{Transformation: aggregate turbidity p95 by sub-basin}}:::xform
    T4{{Transformation: merge grant lines with impact + turbidity}}:::xform

    EMB[/Embedding: text-embed-3-large v2026.03/]:::embed
    R{Retrieval: top-k=8 over FASD bulletins for context}:::retrieve

    I([Inference: claude-opus-4-7 draft memo, T=0.2]):::infer
    TC[(ToolCall: math reconciliation: spent_vs_authorized)]:::tool

    D[[Decision: publish vs. hold-for-review]]:::decide
    O((Output: weekly_memo_2026-05-25.md hash:0xA3F1)):::out

    S1 --> E1 -- "operation: pull;data_shape: rows:412;conf: preserve;det: true" --> T1
    S2 --> E2 -- "operation: snapshot;data_shape: rows:18204;conf: preserve;det: true" -->
T2
    S3 --> E3 -- "operation: window;data_shape: rows:60480;conf: preserve;det: true" --> T3
    T1 -- "join:on=grant_id;conf: preserve;det: true" --> T4
    T2 -- "join:on=sub_basin_id;conf: preserve;det: true" --> T4
    T3 -- "join:on=sub_basin_id;conf: preserve;det: true" --> T4
    T4 -- "summarize:tokens<=1200;conf: downgrade;det: false" --> EMB
    EMB -- "vectorize;conf: preserve;det: false" --> R
    R -- "top-k=8;conf: preserve;det: false" --> I
    I -- "tool_request;conf: preserve;det: true" --> TC
    TC -- "reconcile_result;conf: upgrade-on-merge;det: true" --> I
    I --> D
    D -- "decision_origin: agent OR human-override;det: false" --> O

    classDef source fill:#1f4d3a,color:#fff,stroke:#0d3324;
    classDef extract fill:#2d6a4f,color:#fff,stroke:#1b4332;
    classDef xform fill:#a98467,color:#fff,stroke:#6b4f3a;
    classDef embed fill:#b08968,color:#fff,stroke:#7f5539;
    classDef retrieve fill:#cdb697,color:#000,stroke:#7f5539;
    classDef infer fill:#3a5a40,color:#fff,stroke:#1b4332;
    classDef tool fill:#588157,color:#fff,stroke:#344e41;
    classDef decide fill:#bc6c25,color:#fff,stroke:#6f4518;
    classDef out fill:#0d3b2e,color:#fff,stroke:#0d3324;
```

Drawing Conventions

1. Time flows top-to-bottom. Branches are horizontal.

2. Every Decision node has exactly one decision_origin recorded per C2.

3. Non-deterministic edges (is_deterministic: false) are the re-run risk surface — s3-reproducibility-controls defines how to constrain them.

4. A lineage map is incomplete if any node lacks the edge labels above, or if any leaf is not a Source row from s3-source-inventory-template.

5. Maps are versioned alongside the system. A model swap, retriever change, or new source is a lineage map version bump, not a silent edit.

Anti-patterns

- "Source: internet" — not a source. Either it is on the inventory or it is not in lineage.
- A Transformation that secretly calls an LLM — that is an Inference node; relabel.
- Merging two confidence_band: low inputs into a high output without an explicit, recorded merge rule. Confidence does not laundry.