# DATA LINEAGE MAP

Reusable LPM Knowledge Object · Information Ecology / Platform Structure

Trace source → transform → consume → decide → act → evidence

## Purpose

Use this template to map how critical data moves through the enterprise before it becomes a dashboard, metric, decision, control, workflow, AI model, or agentic action. The goal is to make data movement visible, owned, evidence-backed, and safe for AI scaling.

| Metadata | Value |
|---|---|
| Object type | LPM Knowledge Object |
| Primary LPM layers | Information Ecology, Platform Structure, Governance Architecture, AI Amplification |
| Connected layers | Ownership Map, Decision Architecture, Communication Architecture, Evidence Checklist, Source-of-Truth Map |
| Primary use | Trace how critical data moves from source system to transformation, metric, decision, workflow, AI model, agent, and control evidence. |
| Website use | Downloadable template, workshop guide, AI readiness resource, JSON object for Lapemo ingestion, and future guided skill. |
| Version | 1.0 |
| Owner | LPM / Lapemo |
| Last reviewed | 2026-06-24 |

# 1. Core principles

| Principle | Meaning |
|---|---|
| Lineage is proof, not decoration | A dashboard, AI output, or decision-grade claim is only reliable when the upstream source, transformation path, owner, and freshness are visible. |
| Every data movement needs a reason | Data should not move through integrations, spreadsheets, reports, warehouses, or AI tools without a defined purpose and accountable owner. |
| Source, transform, consume, act | The map must show where data originates, how it changes, who consumes it, and what decisions or actions it drives. |
| AI requires lineage discipline | AI should not retrieve, summarize, recommend, classify, or act on data unless the source, quality, sensitivity, and human review boundary are known. |
| Data ownership and system ownership are not the same | The business owner defines meaning and acceptable use. The technical owner maintains reliability, access, integration, and observability. |
| Stale lineage creates false confidence | Lineage must include refresh cadence, last validation date, transformation logic, confidence rating, and review triggers. |
| Exceptions must be visible | Manual exports, shadow spreadsheets, duplicate dashboards, broken integrations, and undocumented transformations should be treated as operating-model risk. |

# 2. Required fields

| Field | Definition | Required |
|---|---|---|
| Lineage object ID | Unique identifier for the data lineage object, domain, data product, metric, workflow, or AI use case | Yes |
| Business domain | Customer, employee, product, finance, risk, sales, delivery, operations, platform, or AI | Yes |
| Business question / use case | The decision, metric, process, report, AI use case, or control this data supports | Yes |
| Source system | Original system where the data is created or mastered | Yes |
| Source object / table / document | Specific table, API object, file, event, document, record, or data product | Yes |
| Source owner | Business owner accountable for meaning, accuracy, and approved use | Yes |
| Technical owner | System, data platform, integration, or engineering owner accountable for reliability | Yes |
| Data steward | Role accountable for definition, metadata, quality checks, retention, and lifecycle hygiene | Required when material |
| Transformation steps | Joins, calculations, enrichment, cleansing, aggregation, model features, or manual changes applied to the source | Yes |
| Transformation owner | Role accountable for transformation logic and approval | Yes |
| Integration path | API, ETL/ELT, event stream, file transfer, webhook, manual upload, RPA, or agentic workflow | Yes |
| Storage / processing layer | Warehouse, lakehouse, operational store, BI model, vector index, feature store, cache, or knowledge base | Yes |
| Downstream consumers | Dashboards, metrics, apps, teams, controls, workflows, models, agents, vendors, or customers that consume the data | Yes |
| Decision or action supported | What the data helps decide, automate, approve, escalate, report, or govern | Yes |
| Data quality checks | Completeness, accuracy, timeliness, validity, reconciliation, duplicate checks, or exception thresholds | Yes |
| Freshness window | How current the data must be to remain decision-grade or AI-safe | Yes |
| Sensitivity / classification | Internal, confidential, restricted, regulated, customer-sensitive, employee-sensitive, legal-sensitive, or public | Yes |
| Access rule | Who can view, edit, export, query, embed, retrieve, summarize, or automate against the data | Yes |
| AI usage boundary | Whether AI can retrieve, summarize, classify, recommend, transform, update, or act on the data | Yes |
| Human review rule | When a human must validate the data, output, or action before it is used | Required when material |
| Evidence link | Proof source for the lineage path, such as catalog link, data contract, pipeline run, dashboard definition, or control record | Yes |
| Known gaps / exceptions | Manual steps, undocumented transforms, duplicate sources, stale fields, broken ownership, or ungoverned AI consumption | Yes |
| Confidence rating | High, medium, low, provisional, stale, disputed, or retired | Yes |
| Review date | Date the lineage path must be reviewed again | Yes |
| Version | Object version, owner, last reviewed date, and change history | Yes |

# 3. Data lineage stages

| Stage | What it proves | Examples | Owners | Common risk |
|---|---|---|---|---|
| 1. Create / master | Where the data is first created or legally mastered | CRM account, HRIS employee record, policy repository, Jira initiative | Business owner, system owner, data steward | Unclear master, duplicate records, missing owner |
| 2. Capture / ingest | How the data enters the platform or workflow | API, stream, ETL job, form, file upload, vendor feed | Integration owner, platform owner | Manual exports, hidden spreadsheets, undocumented feeds |
| 3. Transform / enrich | How the data is changed, calculated, joined, modeled, or cleaned | Metric formula, semantic model, feature engineering, data cleansing | Data product owner, analytics owner, engineering owner | Unapproved formulas, untested logic, stale joins |
| 4. Store / index | Where transformed data is persisted or made searchable | Warehouse, lakehouse, BI model, vector index, feature store | Platform owner, data owner, security owner | Uncontrolled copies, retention gaps, shadow indexes |

| Stage | What it proves | Examples | Owners | Common risk |
|---|---|---|---|---|
| 5. Consume / interpret | Who uses the data and through what experience | Dashboard, report, app screen, AI assistant, governance review | Consumer owner, product owner, analyst owner | Conflicting dashboards, no context, weak definitions |
| 6. Decide / act | What decision, workflow, automation, or escalation depends on the data | Prioritization, risk review, customer action, agent handoff, control approval | Decision owner, process owner, control owner | AI acts without review, stale data drives decision |
| 7. Evidence / audit | How the organization proves the data path and decision were valid | Data catalog, lineage graph, pipeline log, decision log, evidence checklist | Governance owner, audit owner, source owner | No proof, no replay, no confidence trail |

# 4. Lineage object types

| Type | Lineage path | Examples | Primary owners | Failure mode |
|---|---|---|---|---|
| Metric lineage | Source to formula to dashboard to executive decision | Revenue, cycle time, AI ROI, risk score, adoption rate | Metric owner, data steward, analytics owner | Competing definitions and dashboard drift |
| Customer lineage | Customer master to engagement to action | Account status, contract, support signal, renewal risk | Sales / success owner, CRM owner, data owner | AI outreach based on stale or wrong context |
| Employee lineage | HRIS to access, org design, capacity, and AI workforce planning | Role, manager, team, skills, access, cost center | People owner, HRIS owner, identity owner | Bad routing, access errors, shadow org structure |
| Work lineage | Work intake to prioritization to delivery and outcome | Initiatives, dependencies, blockers, delivery health, benefits | Product owner, PMO owner, work system owner | Hidden work and false delivery status |
| Governance lineage | Policy/control to exception to evidence and audit trail | Risk, control, policy, approval, incident, exception | GRC owner, risk owner, control owner | Unreviewed exception or missing evidence |
| AI model / agent lineage | Training or retrieval data to prompt/model to output to human review | Model features, RAG source, prompt, recommendation, automated action | AI owner, data owner, governance owner | AI output treated as fact without source trace |
| Knowledge lineage | Knowledge object to published artifact to AI retrieval and use | SOP, playbook, framework object, policy, prompt library | Content owner, knowledge steward, AI retrieval owner | Stale documents becoming enterprise memory |

# 5. Template: data lineage map

| Map element | Entry | Guidance |
|---|---|---|
| Lineage ID | [DL-001] | Unique object ID |
| Business domain | [Customer / Employee / Finance / Work / Risk / AI] | Used for routing and governance |
| Business use case | [Decision, metric, dashboard, process, AI use case, control] | Defines why the lineage matters |
| Source system | [System of record] | Original or authoritative source |
| Source object | [Table, API object, data product, document, event, file] | Specific object being consumed |
| Source owner | [Name / role] | Accountable for meaning and approved use |
| Technical owner | [Name / role] | Accountable for system and integration reliability |
| Transformation steps | [Logic, joins, formula, enrichment, model features] | How data changes before use |
| Integration path | [API / ETL / event / file / manual / agent] | How data moves |
| Storage / index | [Warehouse / lakehouse / BI model / vector index / feature store] | Where data persists or is retrieved |
| Downstream consumers | [Dashboards, teams, workflows, models, agents, vendors] | Who or what depends on this path |
| Decision/action supported | [Decision, automation, escalation, report, control] | What the data influences |
| Quality checks | [Freshness, accuracy, reconciliation, completeness, duplicates] | Evidence that path is reliable |
| AI usage boundary | [Retrieve / summarize / recommend / classify / act / blocked] | Defines safe AI behavior |
| Known gaps | [Missing owner, stale data, shadow copy, manual step] | Exception list |
| Review date | [Date] | Next validation date |

# 6. Version for 500+ employee company

| Dimension | Recommended pattern |
|---|---|
| Design intent | Create basic visibility before the company scales into duplicated tools, unowned dashboards, and AI pilots using weak data. |
| Minimum lineage scope | Map the top 10 to 25 critical data paths: customer, employee, finance, delivery, product, risk, and active AI initiatives. |
| Primary systems | CRM, HRIS, work system, finance system, knowledge base, BI tool, and AI pilot workspace. |
| Operating pattern | Quarterly lineage review with business owners, data steward, technology owner, and executive sponsor. |
| AI focus | AI can only summarize or recommend from named sources with human review for customer, employee, financial, risk, or external-facing use. |
| Red flags | Spreadsheet exports, dashboard-only truth, founder-memory definitions, unowned AI prompts, and unclear data freshness. |

- Start with the highest-risk and highest-value data paths, not every table.
- Name owners for the source system, dashboard/metric, and AI consumer before using the data in AI workflows.
- Kill or flag lineage paths that rely only on spreadsheet exports, local knowledge, or chat-based status.

# 7. Version for 5,000+ employee company

| Dimension | Recommended pattern |
|---|---|
| Design intent | Move from local data knowledge to domain-owned lineage that supports cross-functional decisions and scaled AI use cases. |
| Minimum lineage scope | Map critical data products, enterprise metrics, system integrations, decision dashboards, governance controls, and AI model inputs. |
| Primary systems | CRM, HRIS, ERP, work management, service desk, data warehouse/lakehouse, BI semantic layer, GRC, IAM, AI platform. |
| Operating pattern | Domain lineage owners, data product reviews, data catalog entries, quality thresholds, and formal exception paths. |
| AI focus | RAG sources, model features, agent workflows, and AI outputs must reference governed sources, data contracts, and review rules. |
| Red flags | Multiple dashboards for the same metric, undocumented transformations, BI logic outside catalog, and model/agent consumption without lineage. |

- Manage lineage by domain and data product, not by one-off report ownership.
- Require data catalog entries and evidence links for executive metrics, AI use cases, and control reporting.
- Create an exception workflow for duplicate dashboards, disputed sources, and AI consumption without approved lineage.

# 8. Version for 10,000+ employee company

| Dimension | Recommended pattern |
|---|---|
| Design intent | Create enterprise-grade lineage as a control layer for decisions, risk, AI, audit, regulation, and operating-model resilience. |
| Minimum lineage scope | Map all Tier 1 and Tier 2 data domains, regulatory/control data paths, AI-critical datasets, executive metrics, and external reporting feeds. |
| Primary systems | Enterprise data catalog, MDM, data lakehouse, semantic layer, API gateway, event platform, GRC, IAM, model registry, agent orchestration, observability. |
| Operating pattern | Automated lineage capture, lineage control board, domain data councils, evidence packs, policy-as-code checks, and supersession management. |
| AI focus | AI models and agents require retrieval lineage, feature lineage, prompt/output logging, human review policy, model risk tier, and audit replay. |
| Red flags | Federated business units creating conflicting truth, unmanaged data sharing, vendor feeds without ownership, and AI agents acting on low-confidence data. |

- Treat lineage as an enterprise control layer across domains, platforms, AI systems, and regulatory obligations.
- Automate lineage capture where possible, but require human ownership and governance review for material paths.
- Connect lineage to model risk, audit replay, data contracts, access controls, and executive decision records.

# 9. Scoring logic

| Dimension | Score | What good looks like |
|---|---|---|
| Ownership clarity | 0-5 | Every source, transformation, consumer, and decision path has a named owner. |
| Source clarity | 0-5 | The authoritative source and upstream origin are clearly defined. |
| Transformation transparency | 0-5 | Calculation, enrichment, joins, model features, and manual steps are documented and owned. |
| Consumer visibility | 0-5 | Dashboards, decisions, workflows, controls, AI models, and agents using the data are known. |
| Evidence strength | 0-5 | Lineage is supported by catalog links, pipeline logs, definitions, data contracts, and review history. |
| Freshness discipline | 0-5 | Refresh cadence and stale-data triggers are defined and monitored. |
| AI safety boundary | 0-5 | AI retrieval, recommendation, update, and action boundaries are explicit and reviewable. |
| Exception management | 0-5 | Manual steps, duplicates, shadow sources, broken feeds, and disputed lineage have clear escalation paths. |

**Suggested readiness score: average the eight scores, then classify 0-1.9 as Fragile, 2.0-3.4 as Developing, 3.5-4.4 as Governed, and 4.5-5.0 as AI-ready.**

# 10. AI prompts

- Given this data object, identify the source system, transformation steps, downstream consumers, decision dependencies, and missing owners.
- Compare the declared source of truth against actual downstream dashboards, workflows, AI tools, and manual exports. Flag duplicate or conflicting lineage paths.
- Score this lineage path from 0 to 5 across ownership clarity, source clarity, transformation transparency, consumer visibility, evidence strength, freshness discipline, AI safety, and exception management.
- Identify whether this dataset is safe for AI retrieval, summary, recommendation, classification, or autonomous action. Explain the human review rule needed.
- Generate a lineage exception report showing stale data, undocumented transformations, shadow copies, missing owners, weak evidence, and high-risk AI consumers.
- Create a migration plan to move this lineage path from manual documentation to governed Lapemo ingestion.

# 11. Validation rules

- Every lineage object must have one accountable source owner and one technical owner.
- Every transformation must have a documented owner, purpose, and approval state.
- Every downstream AI use must reference an approved source, freshness window, sensitivity rule, and human review boundary.
- Every decision-grade metric must have a named source, formula, steward, quality threshold, and review date.
- No dashboard, model, agent, workflow, or external report should consume data from an unowned or disputed lineage path.
- Manual exports, spreadsheet changes, and shadow copies must be flagged as exceptions unless explicitly approved.
- If data is stale, disputed, low-confidence, or missing evidence, it cannot be used as authoritative truth without owner approval.
- Status must never be encoded only by color; use labels such as approved, provisional, stale, disputed, retired, or exception.

# 12. Lapemo ingestion mapping

| Lapemo object | Fields / entities | Use |
|---|---|---|
| Knowledge object | Data Lineage Map | Canonical reusable artifact for lineage readiness and |

| Lapemo object | Fields / entities | Use |
|---|---|---|
| | | AI-safe data consumption. |
| Ownership object | Source owner, technical owner, steward, transformation owner | Connects lineage to accountability. |
| Information object | Source, transformation, storage, consumer, evidence | Connects lineage to enterprise knowledge and data products. |
| Platform object | Systems, integrations, APIs, pipelines, BI, model registry, vector index | Connects lineage to enterprise systems under control. |
| Decision object | Decision/action supported, evidence link, confidence | Connects data to decisions and outcomes. |
| Governance object | Sensitivity, access, control, exception, review date | Connects lineage to risk, compliance, and auditability. |
| AI object | AI usage boundary, human review rule, output log, agent consumer | Connects data lineage to AI governance and control. |

# 13. Reusable knowledge-object model

This artifact should exist in four synchronized forms: a human-readable guide, a downloadable template, a machine-readable JSON object, and a guided Lapemo skill. The knowledge object should be versioned, reviewed, scored, and connected to system data over time. It should not auto-update silently. Lapemo should flag stale lineage, missing owners, broken evidence, new AI consumers, and high-risk exceptions for human approval.