# The Security Master Is the Moat

*AI agents, open identifiers, and the changing economics of institutional equity data*

**Conrad Gann · BW Macro · 20 November 2026**

Download PDF · Test a historical ticker · Join the change-feed early-access list

If you run quantitative research or data infrastructure at a hedge fund, the buy-versus-build decision is familiar. Major vendors supply broad histories, durable identifiers and decades of corrections. Your own systems supply control over what a security meant on a particular date and how that meaning enters a backtest. Many large firms operate both, paying not only for the feeds but for the people and software needed to reconcile them.

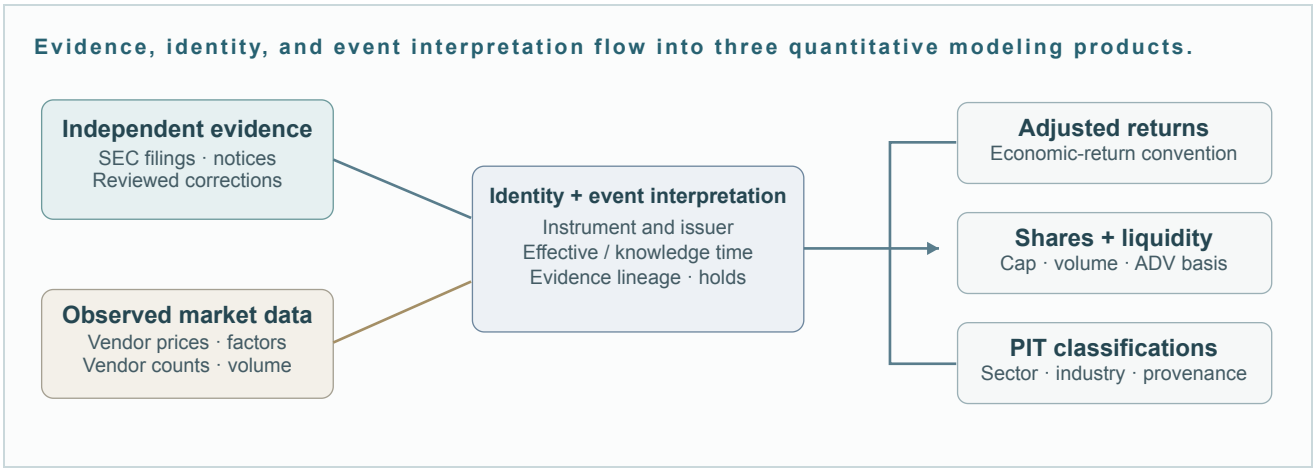
AI agents are beginning to change the cost of that arrangement. Public SEC filings contain much of the evidence required to reconstruct US-equity identity, share counts and corporate events. Open identifiers make it possible to join that evidence without making a proprietary security key the permanent foundation of the system. Coding, filing investigation and exception triage can increasingly be done by a small team directing multiple agents. This does not eliminate licensed market data, or the legal distinctions between an open identifier and a redistributable vendor-supplied crosswalk. It does create new options for firms that want greater control of their data infrastructure.

RiskModels operates a US-equity security master built on FIGI-first identities, regulatory evidence and reviewed corrections. It underpins our production ERM3 risk model today. The master and the code that maintains it remain under active development: the current backlog includes material defects, and resolving those defects will not prevent new corporate events or previously unrecognized errors from arising. The question is how to operate a system that can find, test and correct them without quietly damaging its own history.

We began the repository in October 2025. By 18 November 2026, its main branch held 2,639 commits. Anthropic's Claude was the principal engineer and orchestrator; specialist reviews from other models repeatedly changed our decisions. What emerged was not just a set of ticker mappings but an approach to **historical financial truth**: dated identities, typed corporate events, source-independent evidence, explicit uncertainty and repeatable corrections.

This paper explains the approach through the three outputs quantitative models depend on—adjusted returns, shares and liquidity, and point-in-time classifications—then traces the engineering record that shaped it. It also offers an adversarial test list that can be applied to an internal or licensed master. The engineering account is candid about the difference between what RiskModels runs today and what its event model still needs to become.

## What the model actually needs




For an equity-risk model, a correct ticker-to-issuer join is only a starting condition. Three downstream products make the security master economically consequential.

**Adjusted returns.** The model needs a return series that distinguishes investment performance from changes in units, distributions and securities held. A split is not a windfall; a spin-off creates a new holding; a merger may terminate a position for cash or successor shares. The convention must be explicit. Vendor adjusted-close returns, split-adjusted prices, total-return series and the realized return to a holder are not interchangeable.

**Shares, capitalization and liquidity.** Market capitalization requires price and shares on the same as-traded basis. Share-count histories have to distinguish listed shares, company totals, ordinary shares underlying depositary receipts, and counts restated by a vendor. Volume and average daily volume (ADV) also need a stated adjustment basis. A price and share error can cancel in capitalization while corrupting both inputs and a liquidity measure.

**Point-in-time industry classification.** A historical sector label is a dated judgment, not a permanent attribute of an issuer. A spin-off may create two businesses, a company may change activity, and a taxonomy may be revised. Researchers need to know whether a classification came from a filing available at the time or was retrospectively backcast. That distinction affects historical cohorts, hedge baskets and the interpretation of factor exposures.

ERM3 does not produce all three from one event ledger today. Its published returns begin with the vendor's adjusted-close change and add guards and credits for a limited set of reviewed events. Its shares path is SEC-first but still consumes vendor split factors; ADV uses a vendor-adjusted price/volume basis with incomplete independent checks. Its in-house industry history has dated SEC SIC windows but also backcast classifications. The source and basis of each output therefore matter as much as the output value.

## An identifier does not establish an identity

We chose FIGI first because the identity record needed an open, redistributable key. The production hierarchy prefers `BW-<US composite FIGI>`. If no suitable FIGI is available, it falls back to a CIK-based key, with suffixes to distinguish securities from the same filer; the final fallback is a deterministic opaque serial. The system maintains separate issuer, filer and fund namespaces and dated windows for historical identifiers.

This arrangement addresses a licensing and join-key problem; it does not solve historical identity by itself. A CIK describes a filer, which may have several securities. FIGIs exist at different levels, including composite and share-class identifiers. A valid identifier may refer to the wrong listing context. In one audit, 634 equity rows were keyed to

surviving foreign listings after the intended US line had delisted. A US-venue guard and reviewed re-key proposals followed. A real identifier had supplied the wrong answer to the historical question.

Nor does a `BW–` prefix guarantee permanence. Existing FIGI-form keys survive an ordinary rebuild, while reviewed re-key or merge operations can change or retire an identity. Old-to-new maps exist internally; clients cannot yet retrieve a general public redirect history. A direct-ID path in the resolver was also found to echo a supplied ID as resolved without validating its existence. These are open implementation limits, not features we should conceal behind the description “stable identifier.”

The more durable object is the **decision record**: what instrument was identified, the time interval over which the relationship was believed valid, what evidence supported it, and how a later correction changed that interpretation.

## Corporate events as an accounting problem

Double-entry accounting provides a useful analogy, with limits. An accountant does not retain only the net change in cash; a transaction has structured entries and an audit trail. A corporate action likewise has several effects that should be represented separately: security identity, holder entitlements, shares outstanding, per-unit price, cash consideration and the historical return convention.

Motorola's January 2011 separation illustrates the problem. The company distributed Motorola Mobility shares and executed a one-for-seven reverse split. The split changes the unit count of the continuing parent; the distribution creates a second holding. The vendor represented the combined action with one price factor. Interpreting that factor as a pure share split loses the separate economic legs.

An event ledger would record the transaction's evidence and effective time, then apply **typed transformations** to the resulting holdings and data series. ERM3 already has reviewed event records with input and output legs, exact ratios, deterministic same-day ordering and supersession. It also has an append-only Event Ledger for publication-oriented changes. But the production shares path does not yet consume those reviewed event legs: it continues to use vendor factors. We should not describe this as a completed double-entry system.

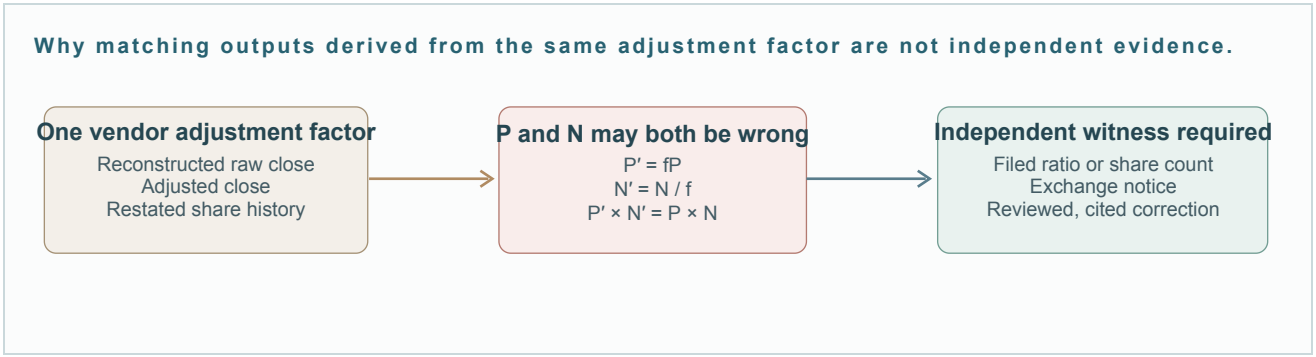
The accounting analogy also identifies a dangerous misconception. A trial balance can agree while transactions are classified incorrectly. Financial-data reconciliation has the same weakness.

Suppose the vendor reconstructs a pre-event price on a mistaken factor $f$, and restates the corresponding share count by the inverse amount. The result may be:

$$P' = fP \quad \cdot \quad N' = N/f \quad \cdot \quad P'N' = PN$$

Capitalization appears correct while both price and shares are wrong. In a November audit, 160 of 3,568 booked split factors were flagged as non-split candidates. Nineteen affected securities were in the top-1,000 universe at their event dates, representing about $0.95 trillion of prior-session capitalization.

The right principle is not that every corporate event conserves value. Issuances introduce new capital; distributions create additional holdings; market prices change for independent reasons. Rather, **each event class needs its own economic relationships, observable consequences and outside witnesses**. Internal agreement among numbers derived from one vendor factor is one witness, however many columns happen to agree.




## How a system can know it is wrong

The SEC filing that states an exchange ratio, the 10-Q cover count, an exchange notice and a vendor-adjusted series differ in what they can establish. Each also has a date on which it became available. For point-in-time work we must distinguish the event's **effective time** from the system's **knowledge time**; a correction learned in November cannot be treated as knowledge available in a backtest five years earlier.

The production record now stores reviewed events with knowledge dates and a `known_as_of` path. That capability is incomplete for other routes: dated aliases can use effective starts as a proxy for knowledge dates, and current master fields still serve some fallbacks. We should therefore describe bitemporal correctness as a requirement being implemented, rather than a universal property of the API.

A concrete identity error shows why an outside witness matters. When ARKO Petroleum began trading as APC in February 2026, the share-count load picked up Anadarko Petroleum's older vendor history, including 491 million shares left open-ended. ARKO's apparent capitalization reached about $7.30 billion. Its own SEC filings supported roughly 46–50 million company shares, and a read-only replay reduced the 9 November estimate to about $710 million. One security ID changed in a 27,819-ID shares build. A later quarantine initially carried forward the already incorrect published history, requiring another correction.

That incident exposes two distinct questions. Was the record assigned to the right identity? And, after an error is discovered, will the release process propagate the corrected history safely? The second question cannot be answered by a ticker resolver alone.

We now distinguish reporting, quarantine and release-stopping findings. The policy changed in November so that eligible single-name errors could be contained without blocking unrelated securities, while shared inputs and systemic failures remained blocking. A quarantine is not a correction; freezing a previously wrong record can preserve the error. The release process therefore needs both containment and a path to adjudication.

## An ensemble of agents, not an autonomous oracle

Claude, principally Opus, built and orchestrated most of the system. We used xAI's Grok—often through Cursor—Moonshot AI's Kimi K3.1 and OpenAI Codex for outside review, with other models on narrower tasks. The division was not ceremonial. Different reviewers repeatedly found different errors, and the ensemble was most useful when one agent was asked to falsify another's interpretation rather than confirm it.

The September identifier-history work is a good example. Run 18 covered 90.11% of in-universe trading days since 2000, passed all 60 acceptance cases and reproduced its row hash. After we removed a restricted research source, a later candidate again passed its checks. Grok's review then found that permissive name matching had attached

unrelated exchange-traded funds, bank notes and other issuers' instruments to common-stock rows. The first guard written in response failed two acceptance cases; more review was needed. Tests had confirmed the examples we knew. The outside review identified examples we did not know to test.

Our assessments of the reviewers were not measured head to head. A bake-off designed on 1 November was never run, and some historic review labels are uncertain. Still, the work record supports a rough, task-dependent ranking:

| Role / operator assessment | Observed strength | Limitation |
|---|---|---|
| **Claude / Opus — principal engineer and orchestrator** | Built the pipelines, coordinated reviews, implemented fixes and maintained continuity across workstreams. Design reviews attributed to Fable pushed the event model from enumerating cases toward explaining how they arise. | Its own implementations and reviews needed external challenge; some early Fable labels actually referred to Opus. |
| **1. Grok — strongest outside reviewer overall** | Quickly found consequential identity and implementation counterexamples, including the false ETF and bank-note matches after acceptance had passed. | Also passed a change later reverted after real-data rehearsal; availability through Cursor was intermittent. |
| **2. Kimi K3.1 — strongest filing reader** | Found the OMEX cumulative split, a depositary-receipt ratio change and a PENN parser duplication during event-model review. | Slower and costlier for general-purpose review. |
| **3. OpenAI Codex — frequent mechanical reviewer** | Especially useful for code-path tracing, tests and replays; showed that two proposed event-model drafts classified none of 66 replay rows. | Missed some economic and filing-specific problems that other reviewers found. |

Gemini contributed to other workflows but is not ranked here: the record does not establish comparable use as an engineering reviewer. These are operator judgments, not evidence that one model generally outperforms another. A controlled comparison should score detection, false alarms, time and cost against the same previously resolved cases.

A consensus of agents is not a consensus of evidence. When five agents reason from one vendor's reconstructed close, adjusted close and restated share series, they may all repeat the same mistake. The useful agent finds a new filing, an overlooked instrument class, a contrary historical observation or a real-data counterexample. The owner then decides whether the finding is strong enough to alter the trusted record. Deterministic code applies accepted transformations; testing and replay establish the consequences. Several apparently good changes were withdrawn at this stage, including a history move that passed two reviews and would have attached an earlier company's price bars to a later issuer.

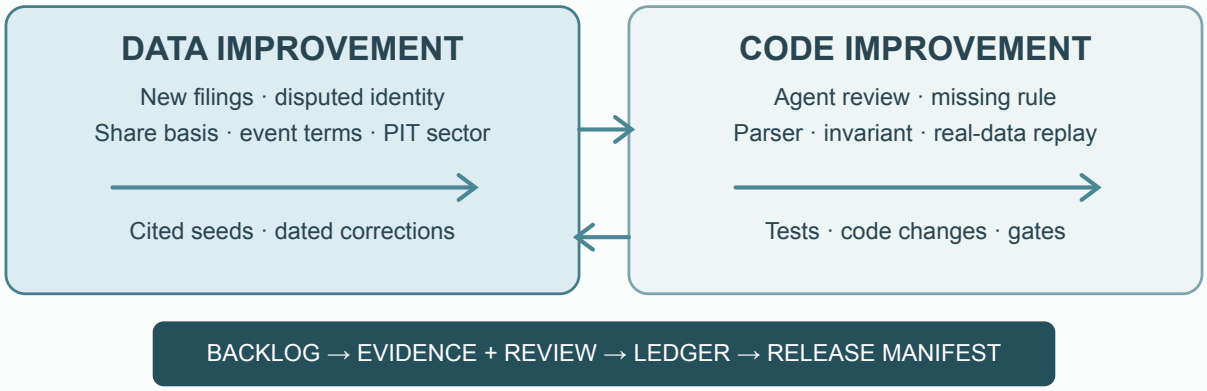
## Two improvement loops, and the meaning of zero state

We came to separate **data improvement** from **code improvement**, although each continually generates work for the other. A data correction changes a dated identity, share basis, event interpretation or classification, ideally with a filing citation and a record of the previous value. A code correction changes a parser, matching rule, transformation, gate or test. A data anomaly can reveal a missing general rule; a new rule can expose a population of incorrect records that had looked internally consistent.

The project's institutional memory is deliberately external to the agents' conversations. `BACKLOG.md` contains open problems classified by impact, cause, state and required action. `LEDGER.md` is append-only and records closed or partially shipped work with a pointer to the PR, commit or decision. A partial solution can enter the ledger while its unfinished remainder stays open. Reviewed seeds retain filing evidence and approval; release manifests record historical changes. This reduces the chance that a new agent reintroduces a rejected fix or mistakes a design note for live code.

We use **zero state** to mean no known actionable code issues remaining. It is a useful target for operating discipline, but should not be confused with zero defects or complete knowledge. Working through known problems has repeatedly uncovered errors that were not visible when those problems were first recorded. The aim is to reduce known material errors while keeping the search for unfamiliar ones active—not to preserve a backlog for its own sake.

**Two connected improvement loops: data corrections and code changes, linked by new discoveries and governed by evidence, reviewed decisions and release tests.**



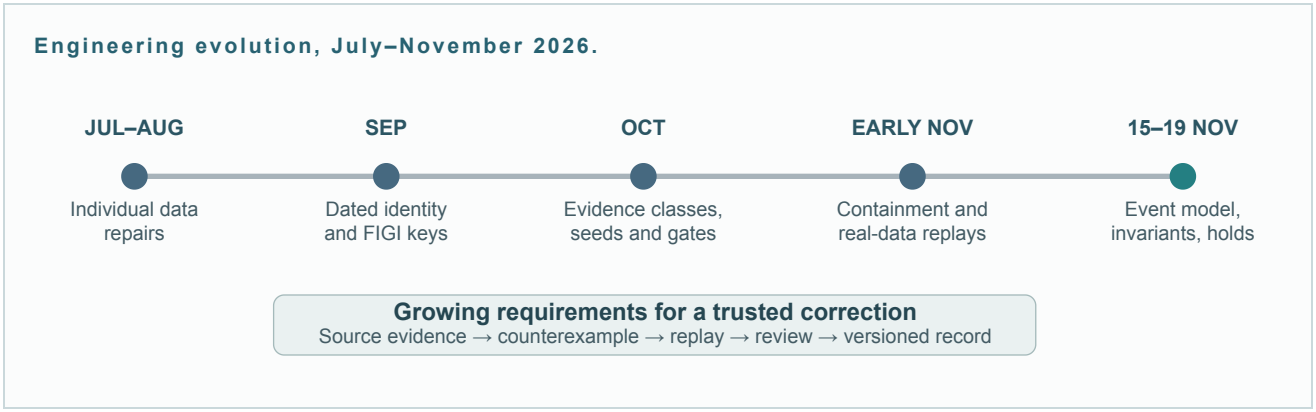

Since 27 October, production publication has used full rebuilds with release manifests. Permanent data corrections are versioned seed records; new identities require proposal and review rather than being minted by the nightly. The thirteen-event framework and the planted tests improved the code's conceptual footing, but the unified corporate-action ledger and its whole-store invariants are not complete. The production split check remains disabled after a full-evidence replay mishandled several compound events. That distinction between the operating system and its next improvement is itself part of the record.

## From isolated fixes to an operating method

The append-only ledger records a progression, not a sequence of finished phases. July and August emphasized stock-specific price, identity and classification repairs. September brought historical identifier windows and the production separation of the licence-free master. October added share-basis classes, reviewed corrections, return-seam adjudication and release manifests. In early November, the team stopped treating every single-name defect as grounds to fail the entire release; eligible defects could be quarantined while shared-input and systemic checks remained blocking.

The mid-November turning point was the discovery that a share-basis fix for one group of stocks could, in replay, create enormous errors for others. The work then moved from issuer-specific patches toward thirteen event classes, thirteen proposed invariants and a generative description of how vendor adjustment factors fold different corporate actions together. The planted test suite ran against frozen real rows, but the full-evidence classifier still failed on compound split-and-spin cases. The check was turned off rather than allowed to publish an unproven interpretation.

The 19 November release record reported 5,416 gate findings, 12 quarantines, zero release-stopping failures and 38 owner holds. That is a statement about the configured release controls, not a measured error rate. The backlog's 18 November count was 39 open class items, ten MATERIAL; additional split-audit and evidence-integration items were being prepared in a review branch. Backlog and ledger entries must be read together to separate a shipped partial fix from its remaining work.




One lesson from this chronology matters to institutions adopting agents: **the correction process has to outlive each agent session**. The value of the ensemble compounds when its discoveries become cited data corrections, reproducible tests, code rules and recorded decisions. Without that memory, fast agents can simply accelerate the rate at which old mistakes are rediscovered.

## The new commercial choice: assemble, test, then decide what to own

Institutional buyers do not need to replace a Tier 1 vendor to benefit from an independent security-identity source. A firm can retain its internal master and licensed prices, use an external resolver to challenge difficult joins, compare filing-backed events against vendor adjustments and decide where its own adjudication remains essential. For a smaller research group, a maintained API can remove more of the infrastructure burden.

Agent-accessible APIs make that incremental approach more practical. An agent can query an identity resolver, retrieve dated risk analytics, compare an answer with the firm's own records and route an unexplained discrepancy for review. This lowers the engineering and procurement cost of an initial experiment; it does not imply that unrestricted market data or a production replacement is available without licenses. A specialist API is valuable precisely because it can be tested alongside existing sources rather than adopted as an all-or-nothing platform.

RiskModels operates the FIGI-first US-equity identity record described in this paper as part of the production **ERM3 risk model** at **https://riskmodels.app**. Programmatic access includes risk analytics and a free identity resolver; some analytics are usage-priced. The resolver's October filing-cited scorecard matched 43 of 61 selected cases, a deliberately investigated sample that is not a population accuracy estimate. Its reviewed-event coverage remains limited, and some fallback and old-ID behaviors still need correction.

The separate **security-master change feed** is in early-access development and has **not yet been delivered to clients**. Its intended scope is filing-cited ticker and name changes, status events, mergers, reorganizations and splits, keyed by FIGI, CIK, RiskModels ID and ticker, with effective and knowledge dates and a correction history. The early-access list is at **https://riskmodels.app/security-master-feed**; participants are offered a specification and sample ahead of the first edition, input into the delivery format and approximately the first year of the feed free.

A serious evaluation should begin with the exception queue, not a demo ticker. The appendix supplies difficult cases, but buyers should also contribute withheld filing-verified cases from their own books and measure false joins, correct abstentions, corporate-action outcomes, point-in-time behavior, revision lineage and manual review hours.

The moat is the maintained capacity to discover that a previous interpretation was wrong, establish a better one from independent evidence and change the published history without losing the reasoning behind the earlier decision. Open identifiers and agents make such infrastructure more attainable. Neither makes the responsibility for its truth disappear.

## Appendix A. Adversarial tests for a historical security master

These are **test specifications**, not claims that RiskModels passes every case. Results should be scored *pass / fail / unresolved / unsupported* on a fixed data vintage, with source citations and a declared return convention.

| Test case | Required behavior |
|---|---|
| BR: Burlington Resources / Broadridge | Separate issuers across ticker reuse; no joined price or share history. |
| KFT: Hungarian Telephone / Kraft | Date-bounded ownership and a correct `unknown` outside documented windows. |
| APC: Anadarko / ARKO | Earlier issuer's bars and 491M share anchor must not attach to the 2026 issuer. |
| CCXI: ChemoCentryx / later SPAC | Old security's data must not enter the new issuer's identity or cap. |
| Motorola, 2011 | Decompose reverse split and Mobility distribution; preserve separate holder legs. |
| Leidos / SAIC, 2013 | Treat simultaneous split and spin-off without turning the compound factor into a pure split. |
| Hilton, 2017 | Handle one reverse split and two distributions on one effective date. |
| Depositary-receipt ratio change folded with other events | Distinguish ADS units from ordinary shares and issuer splits; hold the case when filed ratio terms remain unreviewed. |
| Chevron historical identifiers | Correct dated identifier windows and refusal outside supported windows. |
| Multi-class issuer | Preserve class-specific instruments while distinguishing issuer-wide totals from listed-class shares. |
| Missing Form 25 or missing CIK | Record evidence limits; do not invent a terminal date. |
| A late filing correction | Reproduce both today's best reconstruction and what was known at an earlier cutoff. |
| Vendor-restated price and count | Detect independently when reciprocal errors leave cap unchanged. |
| PIT sector change | Respect dated SIC or reviewed classification evidence; flag backcasts. |
| Merge / re-key of an ID | Preserve a resolvable lineage to the surviving identity, including historical client references. |
| Quarantine after an earlier wrong publish | Ensure the hold does not silently republish the old error as a validated value. |

## Sources and verification

The development chronology, audit figures and implementation descriptions draw on the ERM3 engineering record: dated release histories and manifests, the append-only ledger and open-item backlog, identifier-history review, event-model research, code reviews and filing-cited correction seeds. These internal supporting records can be made available to prospective institutional evaluators on request, subject to licensing and confidentiality constraints. The November split-audit counts are historical findings from a one-off script that has not yet been committed and rerun; they should not be interpreted as a reproducible population-quality estimate. The engineering findings describe the development state through 19 November 2026. The public identity resolver and change-feed early-access page were checked on 23 November 2026; the feed remains in development and has not yet been delivered to clients.

### Selected academic foundations


- Shumway, T. (1997). "The Delisting Bias in CRSP Data." *The Journal of Finance*, 52(1), 327–340. https://doi.org/10.1111/j.1540-6261.1997.tb03818.x
- Jensen, C. S., and Snodgrass, R. T. (1996). "Semantics of Time-Varying Information." *Information Systems*, 21(4), 311–352. https://doi.org/10.1016/0306-4379(96)00017-8
- Buneman, P., Khanna, S., and Tan, W.-C. (2001). "Why and Where: A Characterization of Data Provenance." *International Conference on Database Theory*. (Conceptual reference to provenance, not a source for the ERM3 counts.)


The empirical cases, implementation descriptions, model-review judgments and operational measurements in this paper come from the ERM3 engineering record, not from these publications.

*Analytical infrastructure research, not investment advice.*