Precision Ledger Labs

AI in Accounting Requires Deterministic Financial Architecture

The Financial Systems Maturity Model (FSMM)

Mahesh Sanganeria
Founder, Precision Ledger Labs
April 2026

Executive Summary

"Gartner predicts 60 percent of AI projects will be abandoned by 2026 — not because the technology fails, but because the data isn't ready." [1]

In accounting, the data is never ready.

The reason is structural. As organizations grow, financial systems don't get designed — they get layered. A new payment processor, a new entity, a new currency, a new platform. Each addition makes sense in isolation. Together they create a data environment that no one fully controls and no system fully understands. When artificial intelligence is applied to this kind of environment, it doesn't eliminate the complexity — it amplifies the problem.

Most businesses today run what we call implicit financial systems — manual, error-prone, and dependent on individuals who hold the institutional knowledge required to run them. The alternative is what we call a deterministic financial system — one where data, logic, and transaction states are formally defined, consistently reproducible, and not dependent on any single person. AI belongs on top of that deterministic foundation, not underneath it. This sequencing is not optional.

This paper introduces the Financial Systems Maturity Model (FSMM), which formalizes this progression into three distinct stages: Implicit Financial Systems, Deterministic Financial Systems, and AI-Enabled Financial Systems.

The market is responding to the data readiness problem in different ways — point tools that automate isolated tasks, AI-native ERP platforms, private equity-backed rollups, and large enterprise redesigns. Each serves a segment of the market well. This paper argues that a different model — the AI-native accounting firm, built on deterministic principles — is the right fit for the smaller end of the SMB market: organizations complex enough to need structured financial architecture, but too lean to absorb the cost and friction of enterprise transformation.

The sections that follow define the problem (Sections 1–2), introduce the maturity framework (Section 3), diagnose why the transition fails (Sections 4–5), and describe the organizational model that resolves it (Section 6).

1. The Data Is Already Out of Control

For most growing businesses, the financial data problem doesn't announce itself. It accumulates through entirely normal decisions made under pressure by people doing their best. A customer expands into a new currency — because QuickBooks Online (QBO) requires separate customer records by currency, one person creates the second record by appending the currency to the customer name in parentheses, the next uses a dash, the next uses a space. Three conventions, one customer, no system enforcing consistency. A vendor bill gets approved in haste and coded incorrectly. A two-line customer invoice — one line revenue, one line COGS — gets booked net by one employee and gross by another. The bottom line matches. The P&L does not. In an implicit system, every new ingestion — a new vendor, a new customer, a new bill — is an opportunity for error. Nobody made a bad decision. Nobody was careless. But six months later the books reflect a dozen different people's judgment calls, ingested a dozen different ways, with no common structure connecting them. That is how financial data gets out of control — not dramatically, but incrementally, one reasonable decision at a time.

The signs are familiar to anyone running finance at a growing company. Month-end close stretches longer than it should. Reconciliation depends on one or two people who know where everything lives. Reports get questioned because the numbers don't quite tie. This is not a staffing problem. It is not a tools problem. It is a structural problem. And it will not fix itself.

The research confirms it is widespread. APQC's Open Standards Benchmarking data, drawn from over 10,000 organizations, shows that the slowest companies take twice as long as top performers to complete their monthly consolidated financial statements — ten days versus five [2]. That gap is not explained by company size or complexity. It is explained by architecture. Businesses running on implicit, manually reconciled systems are not just inefficient. They are structurally exposed.

Each of those judgment calls carries a consequence. A miscoded vendor bill creates misstatements. A customer invoice booked net instead of gross distorts the P&L. A naming inconsistency makes automation impossible and reconciliation a manual nightmare — every single period, indefinitely. When the books depend on undocumented judgment calls made by people who may not be there tomorrow, they are not a foundation for decisions, for audits, or for growth.

The cost of losing control is not hypothetical. Revolut, one of Europe's largest fintech companies, experienced a mismatch between its US and European payment systems that resulted in an estimated $20 million net loss over a period of months before the discrepancy was detected. Revolut's auditor, BDO, was subsequently unable to verify approximately two-thirds of the company's 2021 revenues [3]. Implicit systems with weak financial controls are also where fraud hides longest —

the Association of Certified Fraud Examiners found that occupational fraud goes undetected for a median of twelve months, and fragmented, manually reconciled books are the primary enabler [4]. One company, two payment systems, months of undetected losses — and an audit that couldn't sign off on the numbers.

There is now a second reason this matters — one we noted at the outset, and worth repeating. Gartner's data points to the same conclusion — 60 percent of AI projects abandoned, not because the technology fails, but because the data underneath it isn't ready [1]. Informatica's CDO Insights survey reinforces this independently, finding that data quality and readiness is the top obstacle to AI success, cited by 43 percent of organizations [5]. Aaron Levie, CEO of Box, observed the same pattern after meeting with over 20 enterprise AI and IT leaders: "Years of data management fragmentation that wasn't a problem now is an issue for enterprises looking to adopt agents." [6] In accounting, the data is structurally unprepared — not because finance teams are careless, but because the systems were never designed with consistency in mind. Deploying AI on top of fragmented, manually reconciled financial data does not resolve the disorder. It reproduces it at scale.

The question is whether this kind of disorder can be defined precisely enough to measure — and reduced systematically enough to matter. That is the subject of the next section.

2. A Measure of the Chaos

This section develops a framework for quantifying the disorder that accumulates in financial systems. Readers who want only the working definition can carry a single idea forward: entropy is a measure of disorder in a system — and implicit financial systems have a great deal of it. The rest of this section is for those who want to understand how that disorder might be quantified precisely.

The concept of entropy originated in nineteenth-century thermodynamics as a formal measure of disorder in physical systems. Claude Shannon's 1948 reformulation made it portable: entropy measures uncertainty — the more unpredictable the possible states of a system, the higher its entropy [7]. Henri Theil was the first to bring it into accounting, applying Shannon's entropy directly to financial statements [12]. Li, Liang, and Zhang (2024) formalized and empirically validated this approach, demonstrating that higher entropy in balance sheet composition produces stronger stock price reactions and greater analyst attention [13]. This paper applies the same logic one step further: to the close process itself.

Disorder in a financial system is not just felt — it can be measured. Two properties capture it. The first is average close effort: how much work does it take to close the books in a given period? As a company grows, a manual system requires proportionally more people to keep up. An automated system does not. The second is where the close gets stuck. In a well-structured system, the same steps consistently slow things down — bank reconciliation, intercompany elimination, revenue recognition. Teams know what is coming. In a manual system, it shifts every month: one month payroll, the next accruals, the next a prepaid schedule nobody maintained. Together these two properties define a measurable signal of financial system maturity. We do not yet have a precise formula for either — establishing that measure is a tractable direction for future research. The benefits reconciliation cycle illustrates what this looks like in practice.

Consider benefits reconciliation. In most close cycles it ranks outside the top ten — addressed if at all. But when a CFO asks about an unreconciled $500 variance, what ranked outside the top ten suddenly becomes the single most effort-consuming task in the close — consuming $5,000 worth of team effort to resolve. The amount is immaterial to the corporation, the auditor, and the IRS. It is not immaterial to the employee whose FSA deduction left their paycheck but never reached their health account. This is a dual-threshold system: financial materiality and experiential materiality diverge, and the gap between them goes undetected until it triggers an escalation. That shift — from invisible to urgent, from routine to crisis — is what distributional entropy looks like in practice: effort that is reactive, discontinuous, and concentrated precisely where the system was never designed to look.

2.1 Where Entropy Explodes: Blockchain-Based Cross-Border Payments

Blockchain-based cross-border payments is one of the fastest-growing segments in financial services. It is also one of the most architecturally complex: multiple entities, currencies, payment networks, and data formats, scaling faster than most organizations can design for. It makes the entropy problem concrete.

Consider a fintech operating a single corridor — USD settlements into one European currency. The close is manageable: one entity in the general ledger, a few bank accounts, one payment network, one exchange rate source. The company expands into new markets. What follows is not linear growth in complexity — it is exponential. Each new market adds an entity in the general ledger, a few new bank accounts operating under different settlement conventions, a local wallet provider, a regional payment network, a liquidity provider, an exchange rate feed, and an off-ramp provider — each with its own data format, its own timing conventions, and its own definition of a settled transaction. New customers introduce additional currencies, fee structures, and reconciliation touchpoints. None of these systems were designed to speak to each other. Each addition increases average close effort and introduces a new place where the close can get stuck — in the general ledger, the bank reconciliation, or the FX revaluation that nobody can close on time. That is entropy accumulating in real time — not dramatically, but layer by layer, one new system at a time.

The scale of this market makes the example concrete. The Deloitte Center for Financial Services predicts that 1 in 4 large-value international money transfers will settle on blockchain-based platforms by 2030 [14]. The infrastructure is being built now: Mastercard / BVNK ($1.8B, March 2026)[15]; Stripe / Bridge ($1.1B, February 2025)[16]; Rain ($250M, January 2026)[17]; Conduit ($36M, May 2025)[18]; Tazapay ($36M, March 2026)[19]; TransFi ($19M, March 2026)[20]; Levl ($7M, February 2026)[21]; OpenFX ($94M, March 2026)[22]. The infrastructure for a new payments architecture is being built at speed and at scale. The accounting architecture to support it deserves the same attention.

This is the problem the Financial Systems Maturity Model is designed to address.

3. Financial Systems Maturity Model

The Financial Systems Maturity Model (FSMM) describes the desirable evolution of financial architectures across three system states. The diagram below illustrates the progression:

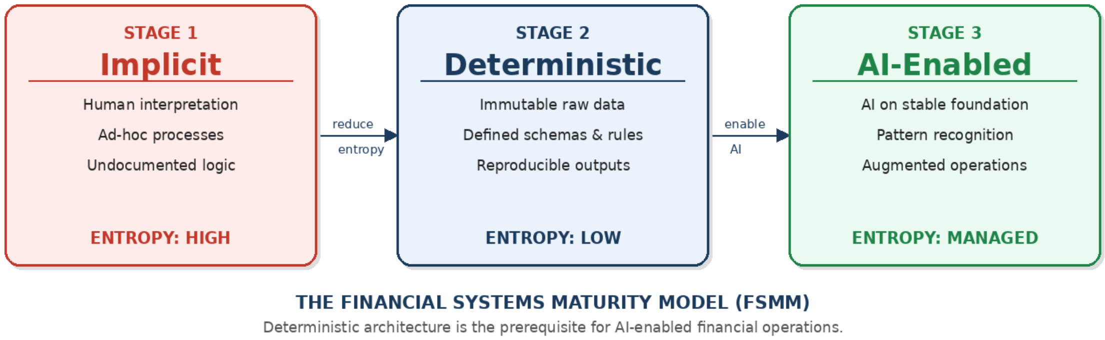

Figure 1: The Financial Systems Maturity Model

3.1 Implicit Financial Systems

Implicit financial systems rely heavily on human interpretation.

Financial workflows are dominated by spreadsheets, manual reconciliation processes, and undocumented transformation logic. Financial outputs are frequently reconstructed through a sequence of manual adjustments rather than derived systematically from well-defined data transformations.

These environments commonly exhibit several structural weaknesses:

- **Manual data replication.** The same financial data may be entered manually in multiple spreadsheet locations, violating the principle of a single source of truth [23].When discrepancies arise, it becomes difficult to determine which value is authoritative or where an error was introduced.
- **Modification of raw operational data.** Raw operational data — such as reports generated by banks, payment processors, or financial platforms — is often edited directly before reconciliation. Instead of preserving raw reports as immutable inputs to structured processing logic, transformation logic becomes embedded within the reports themselves through manual edits, inserted formulas, or ad-hoc adjustments.
- **Hard-coded values.** Key parameters, adjustments, or mapping logic may be embedded directly inside spreadsheet formulas rather than referenced through controlled inputs.

- **Cross-workbook dependencies.** Spreadsheet models frequently reference external workbooks located on individual machines or shared drives. These references may break when files are moved, renamed, or shared with other users.
- **Embedded undocumented logic.** Important reconciliation rules often exist only inside complex spreadsheet formulas or within the personal knowledge of the operators maintaining the model.
- **Cell comments as documentation.** Informal comments embedded within spreadsheet cells frequently serve as the only explanation of how particular calculations or adjustments are performed [24].
- **Formula boundary inconsistency.** In spreadsheet models with large populated ranges, formulas frequently vary from cell to cell at row and column boundaries — particularly at the start and end of data regions. Without clearly defined formula regions and consistent structure that allows formulas to be dragged and repopulated, errors at boundaries are difficult to detect and compound across periods [24][26].

Over time, the reliability of financial outputs becomes dependent on institutional knowledge held by the individuals performing reconciliation. System behavior in implicit financial architectures is therefore non-deterministic and operational entropy is high.

In practice this means extended close cycles, [2] elevated audit risk, and a finance function that scales headcount rather than capability — leaving the organization structurally unable to support the analytical demands of a growing business. The consequences of spreadsheet-based financial architecture are not hypothetical: the US Senate Permanent Subcommittee on Investigations found that a formula error in an Excel risk model — dividing by the sum of two values rather than their average — contributed to $6.2 billion in losses at JPMorgan Chase in 2012 [25].

Multi-entity consolidation illustrates this at scale. When an accounting firm consolidates five or more entities, the standard approach is to add up individual balance sheets, income statements, and cash flow statements in Excel. This is intrinsically wrong: once the P&L and balance sheet are frozen, cash flow should be derived from them — not determined independently by summing individual cash flow statements. The error compounds in multi-currency environments, where translation must be performed at different exchange rates for different statement components, and intercompany balances must net to zero across entities. In practice, the cash balance frequently does not resolve, and the process can consume hours of skilled labor each close cycle — a problem consistent with research showing that approximately 94 percent of spreadsheets contain errors, with error rates in multi-entity consolidation workbooks exceeding 20 percent for groups with ten or more entities [26]. With deterministic automation, the same consolidation can be completed in minutes.

3.2 Deterministic Financial Systems

Deterministic financial systems introduce explicit architectural structure designed to eliminate ambiguity in financial workflows.

In these systems, raw operational data is preserved as immutable input. Data originating from banks, payment processors, enterprise systems, or other financial platforms is stored in its original form and serves as the foundation for all downstream processing. Mapping tables serve as the organizational layer that makes this possible — translating source-specific data formats, account codes, and field names into a consistent internal structure before any transformation logic is applied. Transformation logic is implemented separately from the raw data through defined processing rules rather than embedded directly within reports or spreadsheets.

Financial data flows through clearly defined schemas that specify the structure, meaning, and permissible transformations of each data element. Instead of ad-hoc spreadsheet manipulation, financial workflows operate through rule-based processing engines that apply consistent reconciliation logic to standardized data inputs.

Transaction lifecycles are explicitly modeled through defined state transitions. Each financial event progresses through a predictable sequence of states such as ingestion, classification, reconciliation, exception handling, and final ledger posting. Because these states are formally defined, the status of every transaction within the system can be determined at any point in time.

Exception handling is similarly structured. When transactions cannot be reconciled automatically, they are routed through defined exception workflows rather than resolved through undocumented manual adjustments. These exception processes create a controlled environment in which human judgment can be applied while preserving a clear audit trail of decisions and outcomes.

Under deterministic financial architecture, reconciliation becomes a designed property of the system rather than an activity performed after the fact. Identical inputs processed under identical rules produce identical outputs, allowing financial results to be reproduced consistently.

This reduction in operational entropy provides the stability required for higher levels of financial system maturity.

For finance leaders, this means a faster close, cleaner audits, and financial data they can trust without manually verifying every number [27].

To make this concrete, consider how a prepaid expense schedule is typically managed. In an implicit system, the AP clerk creates a prepaid entry when a vendor bill arrives. If the bill is delayed, has an incorrect period description, or is missing details, the prepaid is missed and corrected in a future period — creating misstatements, rework, and audit risk. The process is entirely reactive: the system only knows about an expense when a human feeds it the right information.

In a deterministic system, the logic is inverted. The system knows a prepaid should exist because it has ingested the signed contract or matched against payment history. It proactively creates the expected prepaid entry, flags the missing or delayed bill as an exception, and routes it for review. The institutional knowledge — that this vendor bills quarterly in advance — moves from the AP clerk's memory into the system architecture itself. Errors become visible immediately rather than surfacing in the next close cycle.

The accrual-reversal cycle illustrates the same class of problem. A legal expense accrued in January is reversed in February — but when the invoice arrives in April, the prior accrual is no longer visible in the workflow. The expense is recorded again. Two periods are now misstated. The problem compounds when different people apply different reversal logic: one reverses the accrual in the following period and codes the invoice to expense; another uses the invoice itself to reverse the accrual by coding to the balance sheet. Both approaches are internally coherent. Together they are incompatible — and when both exist in the same system, the same transaction can be reversed twice. This is not a personnel problem. It is what happens when reconciliation logic lives in people's heads rather than in the system.

3.3 AI-Enabled Financial Systems

AI-Enabled financial systems introduce artificial intelligence operating on deterministic financial infrastructure.

A common question at this point is whether AI can be used to accelerate the transition from implicit to deterministic systems — and the answer is yes. AI tools are increasingly used to generate automation scripts, draft data transformation rules, build reconciliation logic, and document institutional knowledge that previously existed only in people's heads. This is a legitimate and valuable use of AI, and it is how many deterministic systems get built efficiently today. When AI generates a reconciliation script, that script runs deterministically — identical inputs produce identical outputs, every time, without further AI involvement. The AI was present at the moment of construction and then stepped back. What remains is a rule, not a judgment.

AI-Enabled financial systems, as defined in the FSMM, are categorically different. Here, AI is embedded in the running system itself, making continuous operational decisions — classifying transactions, detecting anomalies, flagging exceptions, generating forecasts[28]. That kind of AI cannot operate reliably without a deterministic foundation beneath it, because it is not building the system once — it is running on top of it, every period, on live financial data. When the underlying data is inconsistent or ambiguous, these same models learn and reproduce the disorder rather than resolving it. The sequencing requirement remains: deterministic architecture first, AI orchestration second. The enterprise technology community is reaching the same conclusion independently — Aaron Levie, CEO of Box, has described the requirement as maintaining “some sort of church and state between the deterministic side of your software and the non-deterministic side.” [29]

The consequences of applying AI directly to implicit systems are not abstract. Three concrete examples illustrate how disorder gets encoded rather than resolved.

Inconsistent entity representation. The naming convention problem introduced in Section 1 — the same customer recorded with parentheses, a dash, or a space depending on who created the record — illustrates this directly. To a human reviewer, these are clearly the same entity. To an AI system, they are distinct patterns. When trained on this data, the model may begin to associate each variation with different behaviors or classifications, even though the differences are purely accidental. The inconsistency is not resolved. It is learned.

Reconciliation noise treated as signal. In an implicit system, reconciliation often includes manual adjustments — timing differences, small write-offs, or unexplained variances cleared to force balances to match. When AI is applied to this environment, these adjustments can be interpreted as recurring patterns rather than exceptions. Over time, the model may learn to expect and reproduce these discrepancies, treating what should be investigated as normal system behavior.

Incomplete transaction lifecycles. Financial events in implicit systems are not always tracked through consistent states. A transaction may be recorded, adjusted, partially reconciled, and finalized through a series of undocumented steps. When AI models are trained on such data, they observe fragments of processes rather than complete, well-defined lifecycles. This limits their ability to generalize correctly and increases the likelihood of inconsistent or incorrect outputs when applied to new data.

In each case, the underlying issue is the same. AI does not correct structural disorder. It encodes it. Systems that lack consistent definitions, stable transformation logic, and explicit transaction states provide ambiguous training signals. The result is not improved decision-making, but automated inconsistency.

The practical result, on a deterministic foundation, is a finance function that can do more with the same team — flagging exceptions automatically, forecasting with greater accuracy, and surfacing insights that manual processes would never produce.

Figure 2: FSMM Stage Comparison

	Implicit	Deterministic	AI-Enabled
Data handling	Raw data modified before use; no single source of truth	Raw data preserved as immutable input; all transformations tracked	Clean data fed continuously to AI models with full lineage
Reconciliation	Manual, ad-hoc, dependent on individual knowledge	Rule-based, automated, reproducible with defined exception workflows	AI-assisted matching, anomaly detection, and discrepancy flagging
Audit readiness	Difficult; logic lives in spreadsheets and institutional memory	Full audit trail by design; every transformation is traceable	Continuous audit-readiness; AI generates narrative explanations
Scalability	Headcount scales with transaction volume	Architecture scales; headcount decouples from volume	Capability scales; AI handles routine classification and reporting
Close cycle	Extended; prone to errors and rework	Faster; consistent outputs reduce rework	Continuous close becomes possible
AI readiness	Not possible; AI amplifies existing disorder	Foundation established; AI can be reliably deployed	AI operates on stable, structured financial data

If the progression from implicit to deterministic to AI-Enabled is clear in theory, why do so many organizations fail to make the transition? The next two sections examine the structural traps that prevent it.

4. The Partial Automation Trap

For in-house finance teams, the instinct when processes break down is to automate the part that hurts most. An invoice processing backlog gets an AP automation tool. Reconciliation takes too long and gets a dedicated reconciliation platform. Reporting is slow and gets a BI layer on top. Each decision is reasonable. Each tool, in isolation, works.

The problem is architectural. Each tool arrives with its own data structures, assumptions, and processing logic. None of them were designed to connect to each other, and the implicit financial system underneath them was never designed to connect to anything. The result is automation islands — portions of the workflow that run automatically while everything around them remains manual. The complexity does not disappear. It shifts. Now the outputs of the automated tools have to be reconciled against the manual records, adding a new layer of integration work on top of the original problem. HFS Research's CFO survey identifies fragmented data and the absence of centralized data architecture — not the tools themselves — as the primary barriers to effective F&A automation [30]. The technology works. The foundation does not.

Over time, the finance function accumulates a collection of partially automated subsystems held together by manual processes and institutional knowledge. SSON Research & Analytics found that 88 percent of organizations report only moderate or lower satisfaction with the ROI of their financial automation investments, with fragmented implementations cited as a leading cause [31]. The tools worked. The system did not.

From the FSMM perspective, this is the wrong sequencing. Partial automation attempts to introduce automated processing before deterministic financial architecture exists. Rather than reducing operational entropy, it increases it — more data transformations, more reconciliation points, more undocumented dependencies. Reliable automation requires deterministic architecture as its foundation. Without it, each new tool compounds the fragmentation it was meant to solve.

The partial automation trap explains why technology decisions alone cannot resolve the problem. But even when finance leaders recognize the need for architectural change, a different set of barriers prevents it — not technical, but organizational.

5. Organizational Friction

The transition from implicit financial systems to deterministic architectures rarely fails for technical reasons. The primary barrier is organizational friction. These frictions are not incidental. They are structural properties of how accounting operations are organized. Each friction reinforces the others, creating a self-sustaining equilibrium that resists point interventions.

5.1 Operational Deadline Friction

Accounting operations are governed by recurring deadlines — monthly closes, reporting cycles, audits. Personnel responsible for completing these tasks are reluctant to introduce system changes that could disrupt established workflows, even when those changes would ultimately improve them. Automation initiatives are postponed continuously in favor of completing existing processes. McKinsey’s research on large-scale transformation programs finds that organizations lose an average of 42 percent of expected transformation value in the execution and sustaining phases, precisely where deadline pressure and implementation fatigue take hold [32].

5.2 Defensive Ownership of Existing Processes

Financial workflows evolve gradually and are refined by the teams responsible for operating them. Proposed structural changes are often interpreted as criticism of existing systems rather than improvements to them. This dynamic resembles the well-known “Not Invented Here” (NIH) phenomenon, in which teams are reluctant to adopt ideas developed outside their own group [33].

5.3 Role Preservation and Employment Risk

Automation initiatives generate concerns among employees whose roles involve manual financial processing. Even when automation is intended to augment human work, it is frequently perceived as a threat to job stability. Gartner’s Workforce Change Survey found that employee willingness to support enterprise change dropped from 74 percent in 2016 to 38 percent in 2022 — a collapse driven in part by the accelerating pace of technology-driven workplace transformation [34]. In accounting, where manual processing roles are directly in automation’s path, this resistance is structural rather than incidental. By 2026, the pattern had extended to AI specifically: a global survey of 3,750 executives and employees found that roughly 80 percent of enterprise workers were either avoiding or actively rejecting AI tools their employers had deployed — 54 percent completing work manually instead, 33 percent not using AI at all [35].

5.4 Economic Misalignment

Deterministic financial architecture requires substantial upfront investment in system design, automation development, and process redesign — before any of the long-term cost benefits materialize. Both clients and accounting firms prefer predictable operating costs and typically hesitate to commit to initiatives whose final cost and duration are uncertain. SSON Research & Analytics found that only 33 percent of organizations have strong executive support for automation expansion, with competing priorities and cost justification challenges holding the rest back [31]. Automation initiatives stall even when the long-term economic benefits are widely recognized.

> *These four frictions do not operate independently. Deadline pressure prevents experimentation; process defensiveness blocks redesign; employment concerns slow adoption; and economic misalignment starves investment. Addressing any single friction in isolation leaves the others intact. The structural nature of these barriers is precisely why incremental improvement strategies — introducing better tools, offering training, or phasing in changes — consistently fail to produce architectural transformation. What is required is not a better approach to managing the transition within existing organizations, but a fundamentally different organizational model.*

6. Toward AI-Ready Financial Systems: The AI-Native Accounting Model

The path forward is the AI-native accounting firm — built for AI adoption on deterministic principles, rather than retrofitted toward it through incremental modernization.

The same pattern played out in financial services. When neobanks emerged, they didn't reform legacy infrastructure — they made it irrelevant [36]. The AI-native accounting firm works the same way. Rather than layering automation onto existing processes, it builds operational infrastructure from the ground up: immutable data ingestion pipelines, schema-defined reconciliation engines, explicit transaction state models, and structured audit frameworks as core architectural components rather than compliance add-ons.

Scale the entropy problem across an entire accounting firm — dozens of clients, hundreds of workflows, all running on implicit systems — and the investment case becomes clear. The partnership model cannot self-fund the transformation. Outside capital can.

The traditional partnership model that has governed accounting firms for decades is itself under structural pressure — not from internal reform, but from external capital responding to the AI transformation the partnership model cannot self-fund. Private equity firms have begun acquiring and consolidating accounting practices at an accelerating pace: TowerBrook Capital Partners invested in EisnerAmper in 2021, New Mountain Capital acquired a majority stake in Grant Thornton in 2024, Hellman & Friedman took a significant position in Baker Tilly in 2024, and Blackstone invested in Citrin Cooperman in 2025. As Bloomberg Tax has reported, this wave represents “a reckoning for the partnership model.” [37] The pattern is clear: the partnership structure cannot self-fund the technology transformation required to remain competitive, and outside capital is stepping in to restructure the industry from the top down. This creates both competitive pressure and a market opening for AI-native accounting organizations where deterministic architecture is the starting point, not the destination.

The market is already responding to the need for an AI-native approach. DualEntry, founded in 2024 and backed by Lightspeed Venture Partners and Google Ventures with over $100 million in total funding, has built what it describes as the first ERP platform designed natively with AI — explicitly rejecting the approach of layering AI onto legacy accounting systems [38]. Its co-founder stated the case plainly: “In the past, companies that didn’t shift from on-premise to cloud systems found themselves left behind. Now, the possibility to transition from cloud systems to AI-powered systems is widening the gap even further.” Basis, backed by Accel and Google Ventures at a $1.15 billion valuation, is deploying AI agents across accounting workflows at 30 percent of the top 25 US accounting firms [39]. The AI-native accounting firm is not a future concept. The capital and the companies building it are already in motion.

A platform like DualEntry can simplify the software stack — but every organization still needs an accounting firm to operate it. If that firm runs on implicit processes, the software stack doesn’t matter.

6.1 Platform Economics and Reusable Infrastructure

The most common question at this stage is practical: how does an organization move from Stage 1 to Stage 2 without compromising current operations?

The instinct when moving from Stage 1 to Stage 2 is to automate incrementally — add a tool here, fix a process there. That is the partial automation trap. The distinction is architectural intent. Point tools are introduced reactively — each solving an isolated problem, layered on top of an implicit foundation that was never designed to connect them. In a deterministic transition, a unified architectural plan is defined before implementation begins. What gets built incrementally is not a collection of disconnected tools. It is a single coherent system, delivered in stages.

In practice, the transition cannot compromise ongoing financial operations. Each phase runs in parallel with existing processes until outputs are validated and reconciliation integrity is confirmed. The close cannot be put at risk. Where to start depends on the organization — a fintech company might begin with bank reconciliation, where the impact is immediate; a SaaS company with thousands of customers might begin with revenue recognition, where the complexity is highest and the payoff largest.

The economics of this transition are where the AI-native model diverges most sharply from traditional accounting. In a conventional firm, automation is built per client and stays siloed to that account team. The cost is high, reuse is rare, and organizational friction prevents what works for one client from benefiting another. The risk runs deeper than inefficiency: a bookkeeper managing five clients carries not just their knowledge but their exposure. A capitalization policy from one client bleeds into another — not through carelessness, but through the cognitive load of context-switching without a system to enforce the boundary. In a deterministic system, the rules are coded to the client. The boundary is architectural, not personal.

In an AI-native firm, the starting point is a template — not a blank page. Each template covers approximately 80 percent of a client's needs and consists of seven components: raw data ingestion rules, mapping tables, transformation logic, a configuration sheet that holds all parameters and customization values in one place — eliminating hard-coded logic from the scripts themselves — a set of executable scripts that operationalize the template end-to-end and can be run by a human or an AI agent depending on the organization's FSMM maturity level, a structured output for ERP ingestion, and an ERP confirmation download. Mapping tables are shared across templates and maintained as a single organizational source of truth, auto-updated as client environments change. The remaining customization — the 20 percent — varies by workflow: payroll, prepaid, and accrual schedules may require as little as 5 percent incremental effort; revenue recognition for a complex SaaS business might require 30 percent.

The canonical structure is what makes this work at scale. A client with twenty bank accounts and five different file formats sends raw data that looks entirely different across sources. After passing through the transformation logic, every file looks identical — enriched with additional metadata for retrieval and reconciliation, ready for ERP ingestion as a single structured feed.

The confirmation step closes the loop. After ingestion, the system automatically downloads the relevant balance from the ERP — the prepaid balance, the revenue figure, the bank balance — and compares it against the independent calculation. When a discrepancy is found, the system surfaces both sides of the comparison with the supporting data already assembled. A human makes the judgment call, but on clean structured information rather than having to hunt for it. This is not AI replacing human judgment — it is architecture making human judgment faster and more reliable.

AI plays a specific role in this model during the construction phase — generating ingestion rules, drafting mapping logic, and producing transformation scripts that would otherwise require significant manual effort. Once deployed, the system operates deterministically. The result is a faster path to an integrated financial architecture, at lower cost, lower implementation risk, and faster time to reliability than building each engagement independently from scratch.

7. Conclusion

Artificial intelligence will likely become an important component of financial operations. However, its effectiveness is constrained by the maturity of the financial architectures on which it operates.

The evidence presented in this paper points to a consistent pattern. Reconciliation failures at companies like Revolut demonstrate that financial system disorder can persist undetected for months and ultimately compromise audited results. The blockchain-based payments example illustrates how exponential complexity accumulates in real time as organizations scale — each new market, currency, and payment network adding entropy that no implicit system can absorb. Multi-entity consolidation reveals a deeper architectural flaw: when cash flow is determined independently rather than derived from frozen financial statements, the resulting errors are a property of the architecture itself, not the complexity of the business.

The organizational barriers to resolving this are structural, not incidental. Partial automation compounds fragmentation rather than reducing it, and the incentive misalignments within traditional accounting organizations make architectural transformation consistently difficult to achieve from within. The emergence of AI-native accounting organizations — and the private equity capital now restructuring the industry — signals that the market has recognized what the profession has been slow to address: the partnership model cannot self-fund the technology transformation required.

> *The future of AI in accounting will depend less on advances in machine learning models and more on improvements in financial system design. Organizations that invest in deterministic architecture today are not merely improving their current operations — they are building the foundation on which the next generation of intelligent financial systems will reliably operate.*

The question is no longer whether AI will transform accounting, but whether financial systems will be ready when it does.

1. Gartner. “Lack of AI-Ready Data Puts AI Projects at Risk.” Press release, February 2025. gartner.com/en/newsroom/press-releases/2025-02-26-lack-of-ai-ready-data-puts-ai-projects-at-risk

2. APQC. Open Standards Benchmarking, General Accounting and Reporting research series. apqc.org

3. Financial Times. Reporting on Revolut’s payment system discrepancies, auditor qualifications, and BDO’s inability to verify 2021 revenues. Multiple articles, 2022–2023.

4. Association of Certified Fraud Examiners. “Report to the Nations: 2022 Global Study on Occupational Fraud and Abuse.” acfe.com

5. Informatica. “CDO Insights 2025.” Annual survey of chief data officers. informatica.com

6. Levie, Aaron. Post on X (formerly Twitter), March 19, 2026. Covered by Fast Company: fastcompany.com/91512532

7. Shannon, C.E. "A Mathematical Theory of Communication." Bell System Technical Journal, 27, 1948.

8. Jorgensen, Erik, and Yu-Hsiang Wu. "Effects of Entropy in Real-World Noise on Speech Perception in Listeners with Normal Hearing and Hearing Loss." Journal of the Acoustical Society of America, 154(6), 3627–3643, 2023.

9. Alsaid, Manar, Siva Parvathi Panguluri, and Suliman Hawamdeh. "Combating Misinformation on Social Media Using Social Noise and Social Entropy as a Measure of Uncertainty." Proceedings of the Association for Information Science and Technology, 61, 2024. https://doi.org/10.1002/pra2.1005

10. Stutzer, M. "A Simple Nonparametric Approach to Derivative Security Valuation." Journal of Finance, 51(5), 1996.

11. Arnheim, Rudolf. Entropy and Art: An Essay on Disorder and Order. University of California Press, 1971 (reissued 2010).

12. Theil, Henri. "On the Use of Information Theory Concepts in the Analysis of Financial Statements." Management Science, 15(9), 459–480, 1969.

13. Li, Nan, Pierre Jinghong Liang, and Gaoqing Zhang. "Revisiting Accounting Entropy: an internal summary measure of accounting numbers classification." Working paper, University of Minnesota and Carnegie Mellon University, April 2024.

14. Deloitte Center for Financial Services. "Better, Faster, Cheaper: Multibank Tokenization Networks Could Transform Cross-Border Payments." Deloitte Insights, 2025. deloitte.com/us/en/insights/industry/financial-services/financial-services-industry-predictions/2025/bank-tokenization-global-payments.html

15. Mastercard. "Mastercard to Acquire BVNK to Connect On-Chain Payments and Fiat Rails." Press release, March 17, 2026. investor.mastercard.com/investor-news/investor-news-details/2026/Mastercard-to-Acquire-BVNK-to-Connect-On-Chain-Payments-and-Fiat-Rails/default.aspx

16. Stripe. "Stripe Completes Bridge Acquisition." Press release, February 4, 2025. stripe.com/newsroom/news/stripe-completes-bridge-acquisition

17. Rain. "Rain Raises $250M Series C to Scale Stablecoin-Powered Payments Infrastructure for Global Enterprises." Press release via PR Newswire, January 9, 2026. prnewswire.com/news-releases/rain-raises-250m-series-c-302657084.html

18. Conduit. "Conduit Raises $36 Million Series A to Scale Use of Stablecoins for Cross-Border Payments." Press release via Business Wire, May 28, 2025. businesswire.com/news/home/20250528156066/en/

19. Tazapay. "Tazapay Raises $36M in Total Series B Funding to Scale Next-Generation Payment Rails Globally; Circle Ventures Leads Extension." Press release via PR Newswire, March 26, 2026. prnewswire.com/news-releases/tazapay-raises-36m-in-total-series-b-funding-302725915.html

20. TransFi. "TransFi Inc. Raises $19 Million to Build an Emerging Markets-Focused Global Payments Platform on Stablecoin Rails." Press release via Globe Newswire, March 17, 2026. globenewswire.com/news-release/2026/03/17/3257569/0/en/

21. Levl. "Levl Raises $7 Million Seed Round to Bridge Traditional Banking and Stablecoin Payments." Press release via Business Wire, February 11, 2026. businesswire.com/news/home/20260211668795/en/

22. OpenFX. "OpenFX Raises $94 Million Series A to Scale Cross-Border Money Movement." Press release via Business Wire, March 31, 2026. businesswire.com/news/home/20260330586743/en/

23. Codd, E.F. “A Relational Model of Data for Large Shared Data Banks.” Communications of the ACM, 13(6), 377–387, 1970.

24. European Spreadsheet Risks Interest Group (EuSpRIG). eusprig.org. Active since 2000.

25. US Senate Permanent Subcommittee on Investigations. “JPMorgan Chase Whale Trades: A Case History of Derivatives Risks and Abuses.” March 15, 2013. hsgac.senate.gov

26. Panko, R.R. “What We Know About Spreadsheet Errors.” Journal of Organizational and End User Computing, 10(2), 1998; updated 2008. Findings corroborated by subsequent field audits showing error rates exceeding 20 percent in complex multi-entity consolidation workbooks.

27. Grant Thornton. “From Chaos to Confidence: Rethinking the Financial Close.” February 9, 2026. grantthornton.com/insights/articles/advisory/2026/from-chaos-to-confidence

28. Singla, Alex et al. “The State of AI in 2025: Agents, Innovation, and Transformation.” McKinsey & Company, 2025.

29. Levie, Aaron. Speaking at TechCrunch Disrupt 2025. Reported in: “Box CEO Aaron Levie on how AI is changing the enterprise SaaS landscape.” TechCrunch, October 29, 2025. techcrunch.com/2025/10/29/box-ceo-aaron-levie-on-how-ai-is-changing-the-enterprise-saas-landscape/

30. HFS Research. “Breaking the F&A Gridlock: Finance Leaders Confront Automation Challenges.” HFS Pulse CFO Survey, November 2024. hfsresearch.com

31. SSON Research & Analytics. “The R2R Automation Playbook: From Piecemeal to Integrated – Transforming Financial Close Performance.” Sponsored by Redwood Software, February 2025. sson-analytics.com

32. McKinsey & Company. “How to Implement Transformations for Long-Term Impact.” McKinsey Global Survey, August 2022. mckinsey.com

33. Katz, R. and Allen, T.J. “Investigating the Not Invented Here (NIH) Syndrome.” R&D Management, 1982.

34. Gartner. Workforce Change Survey. Reported in: “Employees Are Losing Patience with Change Initiatives.” Harvard Business Review, May 2023. hbr.org/2023/05/employees-are-losing-patience-with-change-initiatives

35. WalkMe (SAP). “State of Digital Adoption Report — Fifth Annual Edition.” Global survey of 3,750 executives and employees across 14 countries, 2026. walkme.com

36. McKinsey & Company. “Building a Winning AI Neobank.” November 2022. mckinsey.com/industries/financial-services/our-insights/building-a-winning-ai-neobank

37. Bloomberg Tax. Reporting on private equity acquisitions of accounting firms and the restructuring of the traditional partnership model, 2024–2025.

38. DualEntry. “DualEntry Raises $90M Series A to Redefine ERP as AI-Native.” Press release, October 2, 2025. dualentry.com/funding-announcement

39. Basis. Series B funding announcement, February 2026. techfundingnews.com/basis-ai-accounting-100m-1-15b-unicorn

Most organizations do not fail because their finance teams lack capability. They fail because their systems were never designed to produce consistent outcomes at scale. Deterministic financial architecture is not a theoretical construct — it is an implementable system design. Precision Ledger Labs applies the FSMM in practice, building deterministic financial systems for companies where complexity has outgrown traditional accounting workflows.

About Precision Ledger Labs

Precision Ledger Labs LLC specializes in automation-first accounting systems, building deterministic financial infrastructure for organizations navigating the complexity of multi-entity, multi-currency, and high-volume transaction environments. Our approach implements the Financial Systems Maturity Model as a practical methodology for transforming implicit financial systems into AI-ready architectures.