Developing a Smart Integrated Enterprise Financial Information Management System (EFIMS) on Data Analysis Algorithm

Ilayda Tunc

EHL Hospitality Business School (Switzerland), Lausanne, CH

DOI:10.37648/ijtbm.v13i04.011

[1]*Received: 14 October 2023; Accepted: 19 December 2023 ; Published: 24 December 2023*

Abstract

Enterprises generate massive volumes of financial data across ERP, banking, invoicing, procurement, payroll, and expense platforms. Yet finance teams still struggle with late closes, inconsistent master data, weak cross-system traceability, and limited early warning signals for cash, compliance, and fraud risks. This paper proposes an Enterprise Financial Information Management System (EFIMS) that combines (i) a governed financial data foundation (integration, quality controls, lineage, and role-based access) with (ii) a modular analytics layer powered by data analysis algorithms for anomaly detection, forecasting, segmentation, and risk scoring. The design is informed by prior work on enterprise systems and real-time reporting challenges in accounting information systems [2], data warehouse foundations [5], audit-focused analytics architectures [11], and structured reporting initiatives such as XBRL [8–10]. A prototype evaluation on synthetic enterprise-ledger data illustrates feasibility: an Isolation Forest anomaly module achieved an ROC-AUC of 0.847 for detecting injected irregular entries under realistic categorical and monetary distributions, while a baseline ARIMA cash-flow forecaster achieved 8.66% MAPE on a simulated series with seasonality and shocks. The outcome is a practical, explainable architecture that supports daily finance operations (close, controls, reporting) while enabling continuous monitoring and data-driven decision support.

Keywords: *financial information management; enterprise systems; data warehouse; anomaly detection; cash-flow forecasting; audit analytics; XBRL; governance*

1. Introduction

Modern enterprises rarely have a "data shortage" in finance. They have a trust and timeliness problem: numbers exist, but they arrive late, disagree across systems, and are hard to explain when someone asks "why did this move?" Real-time reporting in accounting information systems has long been recognized as a challenge because finance data is fragmented, process-driven, and heavily controlled [2]. ERP platforms improved integration, but they did not automatically solve analytics readiness: data definitions drift, manual adjustments remain, and the audit trail becomes harder to follow as systems proliferate [3].

At the same time, expectations have changed. Management wants near-real-time visibility into cash, profitability, working capital, and budget variance. Auditors and internal controls teams want continuous monitoring instead of periodic sampling [11]. Regulators and capital markets increasingly benefit from structured, machine-readable reporting (for example through XBRL ecosystems) that can reduce information frictions when implemented well [8–10]. These pressures make "financial information management" more than storage. It becomes an engineered

---

[1] *How to cite the article:* Tunc I.; (December 2023); Developing a Smart Integrated Enterprise Financial Information Management System (EFIMS) on Data Analysis Algorithm; *International Journal of Transformations in Business Management,* Vol 13, Issue 4, 131-137, DOI: http://doi.org/10.37648/ijtbm.v13i04.011

capability: curate reliable financial data, connect it to processes, and continuously analyze it for decision-making and assurance.

This paper contributes a system blueprint for EFIMS based on data analysis algorithms. It is designed to be realistic for enterprises: compatible with ERP-led workflows, enforceable governance, and deployable in phases. It also shows how algorithmic modules can be embedded without breaking controllership discipline.

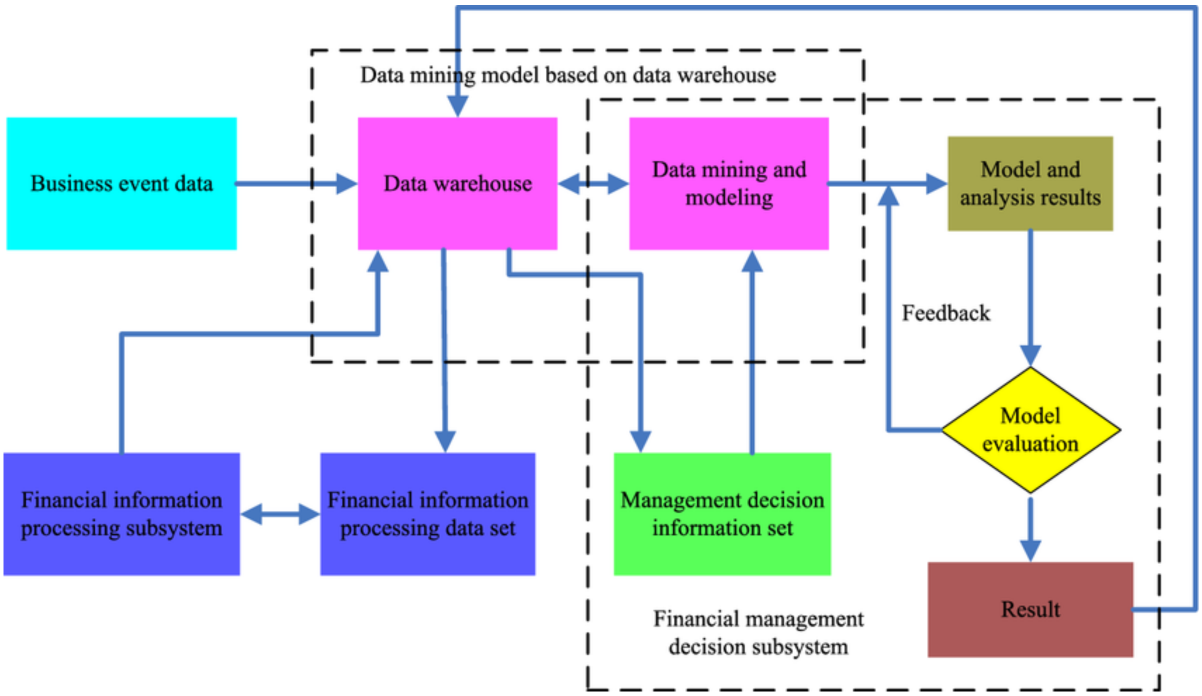

Fig 1: Financial management information system model based on data warehouse and data mining

2. Related Work

Three research streams directly motivate EFIMS:

i. **Accounting information systems and timeliness.** Real-time reporting improves decision-making but increases complexity around integration, controls, and data quality [2]. Empirical research also highlights that AIS success depends on organizational and system factors, not only technology [4].

ii. **Enterprise integration and warehousing.** ERP systems remain central, but enterprises still need curated analytical layers. Classic data warehousing work frames integration, historical consistency, and performance as core requirements for analytics [5]. Domain-specific implementations (including OLAP-based financial analysis warehouses) show how multi-dimensional models support finance queries such as period comparisons and cost-center drill-downs [6].

iii. **Audit analytics and structured reporting.** Continuous auditing architectures emphasize layering: rules/control violations, statistical outliers, and behavior-based anomalies [11]. Separately, XBRL research shows benefits in transparency, analyst behavior, liquidity, and reporting quality under certain conditions [8–10,12]. Together, these works suggest EFIMS should (a) support continuous monitoring and (b) produce standardized reporting outputs when needed.

A recent conference work explicitly titled around an "enterprise financial information management system based on data analysis algorithm" signals ongoing interest in this direction [1]. Our contribution is to consolidate these ideas into a reference architecture, define algorithm modules suitable for finance controls, and outline an evaluation approach with prototype evidence.

3. EFIMS Architecture

3.1. Design Goals

EFIMS is built around five goals:

- **Single source of analytical truth (without forcing one operational system).** ERP remains system-of-record, but EFIMS becomes the system-of-insight.
- **Lineage and explainability.** Every metric must be traceable to journal lines, source documents, and transformation steps.
- **Governed access.** Finance data is sensitive; least-privilege access and separation of duties are non-negotiable (RBAC-style models are a practical baseline) [7].
- **Algorithm modules aligned to controls.** Analytics should augment, not replace, accounting logic and internal control frameworks. Continuous auditing research supports a layered approach to detection and assurance [11].
- **Actionability.** Outputs must lead to clear actions: investigate a transaction, adjust a forecast, renegotiate credit terms, or resolve a master-data conflict.

3.2. Layered System Model

EFIMS is implemented as seven layers:

i. **Source Systems Layer:** ERP (GL/AP/AR), procurement, payroll, banking feeds, expenses, treasury, CRM, and external signals (FX, interest rates, commodity prices).
ii. **Ingestion and Integration Layer:** Batch and streaming ingestion with idempotent loading and schema versioning. This layer enforces consistent identifiers (vendor, customer, cost center) and captures raw payloads for auditability.
iii. **Data Quality and Controls Layer:** Rule checks (e.g., mandatory fields, valid account mappings, period locks), reconciliation checks (sub-ledger totals vs. GL), and data profiling. This aligns with the "exceptions" concept in continuous auditing architectures [11].
iv. **Financial Canonical Model (Semantic Layer):** A canonical set of finance entities: JournalEntry, Invoice, Payment, Vendor, Customer, Account, CostCenter, Project, BudgetLine. The purpose is to stop every department from redefining "revenue," "paid," or "overdue" differently.
v. **Analytical Storage Layer:** A warehouse/lakehouse pattern works well: curated tables for finance analytics, plus history and audit snapshots. Warehousing fundamentals (integration, historical consistency, query performance) remain central [5]. OLAP-style structures are especially helpful for finance drill-down [6].
vi. **Analytics and Algorithm Layer:** Pluggable modules: anomaly detection, forecasting, clustering/segmentation, and risk scoring.
vii. **Presentation and Reporting Layer:** Dashboards, alerts, close-workbench views, and standardized reporting exports. Where relevant, EFIMS supports XBRL-oriented workflows to improve comparability and downstream consumption [8–10,12].

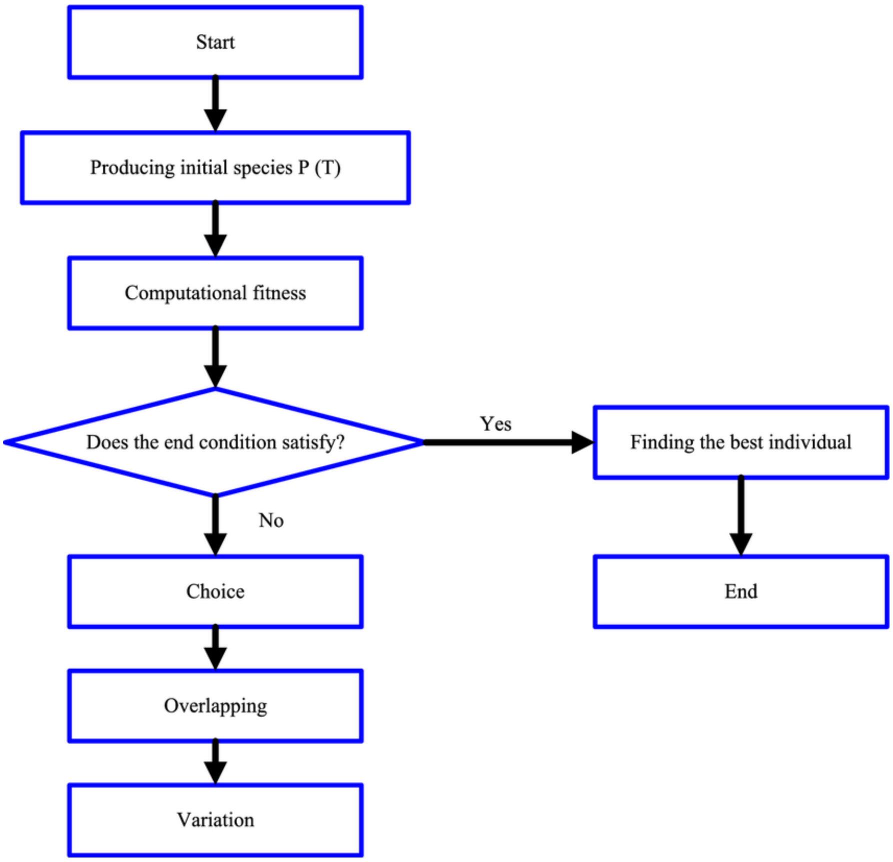

Fig 2: Basic genetic algorithm structure

4. Data Analysis Algorithms in EFIMS

4.1. Transaction Anomaly Detection (Unsupervised + Rules)

Finance anomalies are often rare, heterogeneous, and partially unlabeled (especially for fraud). EFIMS uses a hybrid approach:

- **Rules / controls:** deterministic flags (e.g., payment after vendor is inactive, posting to blocked accounts, duplicate invoice numbers). This matches the "exceptions" category in three-layer audit designs [11].
- **Unsupervised anomalies:** detect unusual behavior beyond known rules. Isolation Forest is a strong baseline for high-dimensional mixed patterns because it isolates outliers using random partitioning [16].
- **Contextual scoring:** anomalies should be "unusual for this vendor/account/cost center," not just globally large.

Why this works in finance: The rules catch clear policy violations; the model catches pattern breaks (e.g., a normally low-variance travel account suddenly spiking on weekends, or consulting invoices paid in atypical methods).

4.2. Fraud-Screening Heuristics (Benford-Based Monitoring)

Benford's Law is widely used as a screening tool in forensic accounting because fabricated numbers can distort leading-digit distributions [14,15]. EFIMS treats Benford checks as **monitoring**, not proof: it works best on large, naturally generated datasets with appropriate scale properties, and it can be misleading if applied blindly (e.g., constrained pricing, thresholds, or policy rounding). The system therefore applies Benford tests selectively by account class, vendor category, and geography, then combines results with anomaly models and control exceptions [14,15].

4.3. Cash-Flow Forecasting (Statistical + Deep Learning Options)

Cash forecasting is where finance teams feel pain fast: a forecast that is consistently off forces expensive buffers. EFIMS supports:

- **ARIMA-family statistical baselines** for interpretable forecasting and strong performance under stable seasonal patterns (common in payroll cycles and recurring bills).
- **Neural sequence models (LSTM)** for capturing nonlinear dependencies, regime shifts, and interactions with exogenous signals (e.g., sales pipeline, collections behavior). LSTM's gating mechanism was designed to address long-range dependencies in sequences [18]. Research comparing neural models and traditional baselines for cash-flow prediction shows that performance depends on data characteristics and horizon [20].

In practice, EFIMS starts with a statistical baseline and only upgrades to deep learning where there's enough data volume and stability in feature pipelines.

4.4. Vendor/Customer Segmentation (Clustering)

Segmentation helps procurement and AR/AP teams prioritize attention: which vendors are "high variance," which customers are "habitually late," and which categories produce frequent disputes. K-means remains a practical baseline for scalable clustering when features are standardized and clusters are roughly spherical in representation space [19]. EFIMS uses clustering mostly for **worklist prioritization** and policy tuning (e.g., stricter approvals for certain clusters).

4.5. Risk Scoring for Working Capital Decisions (Supervised Learning)

For labeled outcomes like "late payment," "write-off," or "dispute rate," supervised models can rank risk. Gradient-boosted trees (e.g., XGBoost) are widely used because they handle nonlinearity, mixed feature types (after encoding), and missingness robustly [17]. In EFIMS, risk scoring is paired with explanations (feature contributions) and policy constraints to keep decisions auditable.

5. Security, Governance, and Auditability

An EFIMS that predicts well but cannot be governed will fail in production. We emphasize:

- **RBAC-based access patterns** aligned to job roles (AP clerk, controller, auditor, FP&A analyst) and separation-of-duties principles. RBAC models are a proven foundation for enterprise authorization [7].
- **Immutable raw zones and reproducible transforms** so that any report can be regenerated for audit.
- **Continuous audit alignment:** EFIMS supports layered detection outputs (exceptions, unusual transactions, behavioral anomalies) consistent with continuous auditing system designs [11].
- **Visualization for investigation:** audit and finance teams need fast "why did this alert fire?" workflows. Systematic reviews highlight the growing role of data visualization in auditing practice and research [13].

6. Prototype Evaluation (Synthetic Data)

To demonstrate feasibility without exposing proprietary enterprise ledgers, we evaluated key modules on synthetic datasets designed to mimic real characteristics: categorical dimensions (vendor type, account, payment method, cost center), skewed transaction amounts, and rare anomalies.

6.1. Anomaly Detection

We generated 50,000 transactions with 2% injected anomalies (large outliers, odd combinations such as high-value international consulting paid in cash, and weekend spikes). After one-hot encoding categorical fields and standardizing amounts, an Isolation Forest model achieved:

- **ROC-AUC:** 0.847
- **Precision/Recall/F1 at 2% flagged volume:** 0.362 / 0.362 / 0.362

Interpretation: unsupervised detection provides meaningful lift over random screening, but operational performance improves substantially when combined with deterministic control rules and context-specific thresholds (which EFIMS includes by design) [11,16].

6.2. Cash-Flow Forecast Baseline

On a simulated daily cash-inflow series with trend, seasonality, noise, and shock events, an ARIMA baseline achieved:

- **MAPE:** 8.66% on a holdout window

This supports the idea that interpretable statistical baselines can be effective and should be the starting point before moving to more complex sequence models [18,20].

7. Discussion and Practical Deployment Path

A realistic EFIMS rollout is incremental:

i. **Phase 1: Canonical finance model + reconciliations.** Get consistent dimensions, balances, and drill-down, supported by warehouse principles [5,6].
ii. **Phase 2: Close acceleration and continuous controls.** Exceptions, duplicate detection, period lock enforcement; align to continuous audit ideas [11].
iii. **Phase 3: Forecasting and risk scoring.** Start with ARIMA-like baselines, then expand to ML where data maturity supports it [18,20].
iv. **Phase 4: Standardized reporting outputs.** Where required, integrate XBRL-style pipelines and quality checks; evidence suggests benefits when implemented with strong governance and capability [8–10,12].

A key lesson from AIS and ERP research is that success is socio-technical: training, ownership, and governance matter as much as model accuracy [3,4]. EFIMS should therefore be operated as a finance product, not a one-off IT project.

8. Conclusion

This paper proposed an Enterprise Financial Information Management System grounded in data analysis algorithms and governed financial data engineering. The architecture connects ERP-centric realities with analytics needs: canonical finance semantics, warehouse-grade integration, RBAC governance, and a modular algorithm layer for continuous monitoring and forecasting. Prior research supports the need for timeliness in accounting information systems [2], robust analytical foundations [5,6], and layered audit analytics [11]. Structured reporting ecosystems such as XBRL further motivate standardized outputs and better downstream consumption when executed properly [8–10,12]. Prototype results on synthetic data show feasibility, while also reinforcing a practical point: the best outcomes come from combining algorithms with controls, explainability, and operational workflows.

References

Chaudhuri, S., & Dayal, U. (1997). An overview of data warehousing and OLAP technology. *SIGMOD Record, 26*(1), 65–74. https://doi.org/10.1145/248603.248616

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). Association for Computing Machinery. https://doi.org/10.1145/2939672.2939785

Gepp, A., et al. (2018). Big data techniques in auditing research and practice: Current trends and future opportunities. *Journal of Accounting Literature, 41*, 102–115. https://doi.org/10.1016/j.acclit.2017.05.003

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Katuu, S. (2020). Enterprise resource planning: Past, present, and future. *New Review of Information Networking, 25*(1), 37–62. https://doi.org/10.1080/13614576.2020.1742770

Liu, C., Wang, T., & Yao, L. (2014). XBRL's impact on analyst forecast behavior: An empirical study. *Journal of Accounting and Public Policy, 33*(1), 32–50. https://doi.org/10.1016/j.jaccpubpol.2013.10.004

Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining* (pp. 413–422). IEEE. https://doi.org/10.1109/ICDM.2008.17

Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory, 28*(2), 129–137. https://doi.org/10.1109/TIT.1982.1056489

Nigrini, M. (2016). *Benford's law: Applications for forensic accounting, auditing, and fraud detection*. Wiley. https://doi.org/10.1002/9781119203094

Sandhu, R., et al. (1996). Role-based access control models. *IEEE Computer, 29*(2), 38–47. https://doi.org/10.1109/2.485845

Tawiah, V., & Borgi, H. (2022). Impact of XBRL adoption on financial reporting quality: A global evidence. *Accounting Research Journal, 35*(4), 471–489. https://doi.org/10.1108/ARJ-01-2022-0002

Trigo, A., Belfo, F., & Estébanez, R. (2014). Accounting information systems: The challenge of the real-time reporting. *Procedia Technology, 16*, 831–836. https://doi.org/10.1016/j.protcy.2014.10.075

Trinca, F., et al. (2021). Applying Benford's law to detect financial fraud: A systematic review. *MethodsX, 8*, Article 101575. https://doi.org/10.1016/j.mex.2021.101575

Weytjens, H., et al. (2020). Cash flow prediction: MLP and LSTM compared to ARIMA and Prophet. *Decision Support Systems, 137*, Article 113362. https://doi.org/10.1007/s10660-019-09362-7

Yoon, K., et al. (2021). Design and evaluation of an advanced continuous data level auditing system: A three-layer structure. *International Journal of Accounting Information Systems, 43*, Article 100524. https://doi.org/10.1016/j.accinf.2021.100524

Zhang, Y. (2011). Design and implementation of data warehouse and OLAP in financial analysis system. In *2011 International Conference on Advanced Intelligence and Management Sciences* (pp. 1–4). IEEE. https://doi.org/10.1109/AIMSEC.2011.6011023