

Enterprise KPI Architecture for Operational Decision Systems: Moving From Reporting Metrics to Decision Intelligence

Sushen Sarkar ¹

¹University of Chicago — Booth School of Business, Chicago, IL, USA
ORCID - [0009-0009-9787-185X](https://orcid.org/0009-0009-9787-185X)

Article Info

Article history:

Received 2026-07-09

Revised 2026-09-24

Accepted 2026-10-02

Keyword:

*Enterprise KPI Architecture,
Operational Decision Intelligence,
Metric Governance Framework,
Performance Measurement
Systems,
Design-Science Research,
Data-Driven Decision Support.*

ABSTRACT

Conventional performance measurement systems in modern enterprises remain largely retrospective, making them inadequate for the speed and complexity of contemporary operational decision-making. This study introduces Enterprise KPI Architecture, a four-layer framework comprising data foundations, metric governance, insight generation, and decision interfaces. Employing a Design-Science Research (DSR) approach, the study synthesizes literature iteratively across five key domains. Three structural KPI failure classes are identified through analytical decomposition, and the framework is evaluated against four DSR-grounded criteria, alongside illustrative composite deployments in logistics and discrete manufacturing. The results show that structural failures—such as metric proliferation, definitional inconsistency, and retrospective orientation—can be effectively addressed through deliberate architectural design. Aligning governed metrics with diagnostic and predictive analytical capabilities delivered via role-specific interfaces reduces decision latency and enhances cross-functional alignment. In conclusion, the gap between data availability and operational effectiveness is an architectural challenge with a structured solution. Organizations with disciplined data and governance foundations are best positioned to adopt emerging AI-driven analytical capabilities. Future research should empirically validate this framework across diverse industry sectors.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

I. INTRODUCTION

Operational data volumes have grown faster than the organizational capacity to act on them. Cloud data platforms, event-driven architectures, and distributed sensor networks have removed most technical barriers to data collection and storage. What persists is a structural gap: the mechanisms designed to translate data into decisions have not kept pace. Traditional KPI systems were built for a different operating environment. They were designed to report what happened, to provide accountability across business units, and to support strategic reviews conducted on weekly or monthly cycles [1]. Those requirements remain relevant. Modern operational environments, however, demand something additional. Decisions about fulfillment routing, capacity allocation, service quality interventions, and demand response must occur within compressed time windows [2]. When the performance measurement system is oriented toward

historical summaries, decision-makers are left interpreting data manually, under time pressure, and without the diagnostic context needed to act with confidence.

The central research question this article addresses is: how should performance measurement systems be architecturally designed to support operational decision-making, rather than simply reporting outcomes? Answering that question requires moving beyond incremental improvements to existing reporting systems. This article proposes Enterprise KPI Architecture, defined here as a four-layer architectural model that governs the structural pathway from raw operational data to operational decision, comprising data foundations, metric governance, insight generation, and decision interfaces, as the structural response to that problem, organizing the full stack from raw data to operational decision into four coherent layers. It also examines the practical barriers that prevent organizations from realizing this transition and identifies the technical and organizational capabilities required to sustain it.

The proposed framework builds on and extends four established research traditions: the Balanced Scorecard, Business Intelligence architecture, the analytics maturity and capability literature, and decision intelligence theory. Each addresses part of the design problem this article targets, but none, in isolation, addresses the full architectural pathway from raw data to operational decision. Section IV D positions the framework against each of these traditions in detail.

A. Research Contributions

This article contributes five specific advances over existing literature in performance measurement and operational decision support architecture:

- 1) Introduces the Enterprise KPI Architecture, a four-layer architectural model that organizes data foundations, metric governance, insight generation, and decision interfaces into a single coherent design discipline.
- 2) Separates KPI governance from analytics, establishing metric governance as a distinct architectural layer rather than an embedded function of data engineering or business intelligence tooling.
- 3) Defines decision interfaces as a first-class architectural layer, with explicit role calibration requirements rather than treating dashboards as undifferentiated visualization products.
- 4) Integrates Decision Intelligence into KPI architecture, operationalizing the data to decision pathway within a governed metric stack rather than as a standalone analytical capability.
- 5) Introduces the governance balance between enterprise standardization and business unit flexibility, through a two-tier model that distinguishes enterprise governed core metrics from unit governed supplementary metrics.

II. METHOD

This article employs a design-science research (DSR) approach, following the construct-evaluate-communicate cycle as described by Hevner et al. [20]. Design-science research is the appropriate methodology for developing and evaluating prescriptive artifacts, in this case an architectural framework, whose value lies in their capacity to solve a class of identified organizational design problems. No primary empirical data collection was conducted; the evaluation standard is internal consistency, literature-grounded design requirement satisfaction, and illustrative plausibility across deployment contexts. The literature synthesis follows an iterative structured review approach, not a formal systematic or scoping review, using forward and backward citation tracing from anchor references in each of the five identified domains, followed by thematic coding of design requirements and failure modes. This approach was selected because the research objective is framework construction rather than exhaustive literature enumeration. The methodology proceeds through three stages.

The first stage is a structured literature synthesis spanning five domains: data warehouse architecture and integration [16]; data governance theory [17]; performance measurement

and KPI system design [7, 8, 11, 12]; analytical methods including statistical process control, anomaly detection, and causal inference [18, 19]; and decision support system and decision intelligence design [2, 4, 5, 10, 13]. Sources were selected on the basis of direct relevance to the identified design problems and citation influence within their respective domains. The synthesis was conducted iteratively: initial source identification through forward and backward citation tracing from anchor references in each domain, followed by thematic coding of design requirements and failure modes.

The second stage is analytical decomposition of identified failure modes in traditional KPI systems. Three structural deficiencies were derived from the literature: metric proliferation and analytical fragmentation, definitional inconsistency across business units, and retrospective orientation without diagnostic capability [8, 9, 11, 12]. These deficiencies were treated as design requirements that any viable architectural alternative must satisfy. The distinction between architectural failures and those arising from organizational culture or data governance maturity is maintained throughout: architectural failures are those embedded in the structural design of the system itself, not in the behaviors of the people using it, and they are therefore not addressable through cultural change alone.

The third stage is framework construction. Each of the four proposed layers was designed to resolve a specific class of identified failure. The data foundations layer addresses the data integration and timeliness deficiencies that undermine metric accuracy. The metric governance layer addresses definitional inconsistency and proliferation. The insight generation layer addresses retrospective orientation by introducing diagnostic and predictive analytical capabilities. The decision interfaces layer addresses the mismatch between generic dashboard outputs and the role-specific information requirements of operational decision-makers. The layer sequence is not arbitrary: each layer depends on those beneath it, and the design requirements cascade accordingly. The framework was evaluated against four criteria derived from DSR practice: internal consistency of layer definitions and dependencies; coverage of the identified failure modes; alignment with the foundational literature in each domain; and illustrative plausibility across two distinct deployment contexts, as presented in Section III D. The four-layer structure was not assumed in advance. Initial decomposition of identified failure classes produced three distinct design requirements: resolving data integration and timeliness deficiencies, resolving definitional inconsistency and proliferation, and resolving retrospective orientation. A fourth requirement emerged from the decision support literature: the mismatch between generic visualization outputs and role-specific decision information needs. Each of these requirements maps to one layer; the number of layers is therefore an artifact of the decomposition, not a prior design choice.

III. RESULTS

A. Limitations of Traditional KPI Systems

Before defining the architectural alternative, it is necessary to characterize the structural deficiencies of conventional KPI systems with precision. These problems are not superficial and are not primarily attributable to poor data quality or organizational culture, though both can compound them. They are embedded in the design assumptions on which those systems were built and cannot be resolved through incremental improvements to dashboards or reporting tools [8].

1) Metric Proliferation and Analytical Fragmentation

Dashboard environments accumulate metrics over time because adding a new indicator is technically inexpensive. Product, operations, finance, and customer experience teams each introduce metrics that reflect their priorities and measurement conventions. The result is an environment in which decision-makers are confronted with dozens or hundreds of indicators, with no structural guidance for determining which signals carry the greatest operational relevance [10]. When the signal-to-noise ratio of a dashboard degrades, the cognitive load on analysts and operational managers increases substantially [2]. High-stakes decisions begin to depend on individual judgment about which metrics to trust and which to discount. Variability in those judgments produces inconsistent operational responses.

Performance measurement systems were historically designed to answer strategic questions over long planning horizons [7]. As balanced scorecard frameworks were operationalized at lower levels of the organization, the discipline around indicator selection frequently dissolved. New metrics were added reactively, in response to specific incidents or stakeholder requests, without retiring obsolete indicators [11]. The accumulation effect is measurable: Brynjolfsson and McElheran document that data-driven decision-making adoption is frequently constrained not by data scarcity but by the organizational inability to identify which metrics carry decision-relevant signal [10]. Parmenter finds that most KPI programs include a large number of indicators that are reviewed infrequently or not at all, and attributes this directly to the absence of a structured indicator retirement mechanism [12]. This is an architectural failure, not a behavioral one. The system provides no mechanism for enforcing indicator discipline because it was never designed to do so.

2) Definitional Inconsistency Across Business Units

Metrics with shared names frequently carry different definitions across departments. An efficiency metric calculated by logistics operations and the same metric calculated by financial planning may differ in their time boundaries, aggregation methods, or source systems. Defect rates, customer satisfaction scores, and throughput measures are particularly susceptible to this form of divergence [9]. Krishnan and Kellner demonstrate that process inconsistency

in measurement directly produces outcome variability, even when the underlying process being measured is stable [9]. Bourne et al. document that definitional misalignment across business units is one of the most frequently cited reasons for KPI program abandonment in organizations with otherwise capable analytics infrastructure [11]. The practical consequence is that cross-functional conversations about performance become contested rather than productive.

Trust in analytics systems erodes not because the systems are inaccurate, but because they produce outputs that are inconsistently interpretable across teams [6]. When users encounter conflicting numbers from different reports, they frequently revert to manual processes or informal information sources they regard as more reliable. The investment in analytics infrastructure fails to produce behavioral change because that infrastructure has not been designed to generate consistent outputs [1]. Definitional governance is the prerequisite for that consistency, and its absence is one of the most common structural failures in conventional KPI programs. This failure is architectural in origin: without a governed canonical definition layer, the system has no mechanism for enforcing definitional consistency even when business units are willing to align.

3) Retrospective Orientation and Diagnostic Absence

The dominant design pattern for KPI systems is backward-looking. Dashboards summarize what occurred in a prior period, and business review processes evaluate performance against historical targets. These summaries are necessary for accountability and strategic planning [7], but they fail to support the operational requirement for understanding why performance is changing and what interventions are available [13]. A customer satisfaction score that has declined by a measurable amount tells an operational manager that a problem exists. It provides no diagnostic pathway to the contributing factors, whether delayed fulfillment, service interaction failures, or product quality variation.

Without that causal linkage, the manager must construct a hypothesis independently, a process that introduces both delay and inconsistency [19]. The absence of diagnostic structure in traditional KPI systems is not an accident of poor implementation. It reflects the original purpose of those systems: to document outcomes and support accountability, not to guide operational response [8]. Table I summarizes the distinguishing characteristics that separate traditional KPI reporting from the architectural model introduced in subsequent sections.

TABLE I
COMPARATIVE CHARACTERISTICS: TRADITIONAL KPI SYSTEMS VS.
ENTERPRISE KPI ARCHITECTURE [7, 10, 11]

Characteristic	Traditional KPI Systems	Enterprise KPI Architecture
Primary purpose	Historical performance reporting	Operational decision support
Metric scope	Functional and siloed	Cross-functional and integrated
Data timeliness	Periodic (daily/weekly/monthly)	Near real-time streaming

Analytical depth	Descriptive summaries	Diagnostic and predictive models
Governance model	Informal or ad hoc	Structured ownership and versioning
Decision interface	Generic dashboards	Role-specific control interfaces
Insight delivery	Analyst-mediated	Automated anomaly and pattern detection

Table I characterizes the two models in descriptive terms. To make the comparative strength of each model explicit, Table II summarizes the same dimensions in a checklist format, indicating which model exhibits the stronger capability profile for each criterion.

TABLE II
COMPARATIVE CHARACTERISTICS: TRADITIONAL KPI SYSTEMS VS.
ENTERPRISE KPI ARCHITECTURE [7, 10, 11]

Capability	Traditional KPI Systems	Enterprise KPI Architecture
Real time decision support	X	✓
Cross functional metric consistency	X	✓
Diagnostic and predictive analysis	X	✓
Structured metric governance	X	✓
Role specific decision interfaces	X	✓
Automated anomaly detection	●	✓
Strategic accountability reporting	✓	✓
Architectural scalability	●	✓

B. Enterprise KPI Architecture: Structural Design

Enterprise KPI Architecture organizes performance measurement into four interconnected layers. Each layer addresses a distinct set of design problems, has defined boundaries and internal functions, and depends on the layers beneath it. The architecture is not a technology stack; it is a design discipline that imposes structure on the relationships between data, metrics, analysis, and decision support [5]. The layers are: Data Foundations, Metric Governance, Insight Generation, and Decision Interfaces. Figure 1 depicts the data-to-decision flow traversal across all four layers. The framework's differentiation from the Balanced Scorecard, Enterprise Performance Management, Business Intelligence Architecture, and Decision Intelligence frameworks is addressed in full in Section IV D. The key distinction is that none of the four existing traditions, individually, addresses the complete architectural pathway from raw data ingestion through governed metric computation to role-calibrated operational decision support; the proposed framework integrates these traditions into a unified layered model.

Each layer has defined input conditions, output products, and dependencies. The Data Foundations layer receives heterogeneous source data and produces a quality-governed, timeliness-matched analytical substrate; its primary limitation

is that data integration completeness and latency targets must be negotiated against legacy system constraints. The Metric Governance layer receives that substrate and produces a versioned, ownership-assigned metric registry; its primary limitation is organizational adoption, canonical definitions have no effect if business units continue computing their own local variants. The Insight Generation layer receives governed metric streams and produces diagnostic and predictive analytical outputs; its primary limitation is model interpretability, specifically whether outputs are explainable enough to support operational action. The Decision Interfaces layer receives pre-interpreted outputs and delivers them through role-calibrated interfaces; its primary limitation is the risk of interface proliferation if role specificity is not bounded by a defined tier structure. The layers are sequentially dependent: a governance failure at layer two propagates as definitional inconsistency into layer three, and an interpretability failure at layer three produces undifferentiated signal at layer four.

A single running example illustrates how a signal traverses all four layers. Consider an order delay flagged in a logistics operation. At the Data Foundations layer, the delay first becomes visible as a timestamp discrepancy between the order management system and the carrier tracking feed, integrated into the unified analytical substrate described below. At the Metric Governance layer, that discrepancy is evaluated against the canonical definition of on time fulfillment, ensuring the delay is measured consistently regardless of which business unit is reviewing it. At the Insight Generation layer, anomaly detection flags the delay as a deviation from the expected fulfillment baseline, and root cause analysis traces it to a specific upstream driver, such as a carrier capacity shortfall, rather than leaving the deviation unexplained. At the Decision Interfaces layer, that diagnosis is delivered to the frontline logistics manager as a role calibrated alert with a recommended rerouting action, rather than as a raw data point buried in a generic dashboard. This single example is referenced again at each layer below to keep the architectural pathway concrete.

The translation mechanism between these two layers operates through three structural elements: pre-filtering, annotation, and escalation routing. Pre-filtering ensures that only signals that meet relevance and confidence thresholds for a given role are surfaced; raw metric streams do not pass through. Annotation attaches the causal context from the Insight Generation layer directly to the alert or recommendation, a logistics manager does not receive a fulfillment deviation in isolation, but paired with the upstream driver identified by root cause analysis. Escalation routing ensures that signals requiring authority beyond the receiving role are automatically forwarded through the tier structure defined in Figure 3, rather than leaving the decision-maker to manually determine whether to escalate.

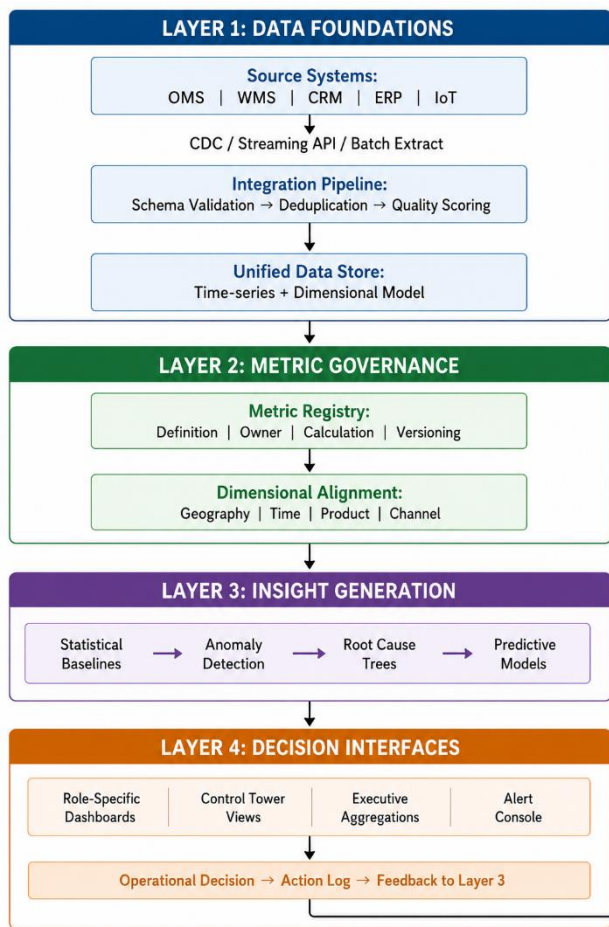


Figure 1. Data-to-Decision Flow Model: Enterprise KPI Architecture Layer Traversal [2, 4, 16, 17]

1) Data Foundations

The Data Foundations layer establishes the shared operational data environment on which consistent metric computation depends. Its boundary spans the ingestion of data from heterogeneous source systems and the production of a unified, quality-governed analytical data substrate. Data originates from order management, logistics tracking, customer relationship management, financial ledgers, and IoT telematics systems. Each of these sources has its own schema, refresh rate and ownership regime, and the resulting metrics will share any discrepancies [16].

Data integration pipelines layer source data into a common operational layer. This is typically responsible for schema alignment, deduplication, late-arriving events, and referential integrity. Schema drift in upstream systems where a source application changes its data structures can cause silent corruption of metric calculations without this schema validation and alerting in the integration layer [16]. The data warehouse architecture provides the subject-oriented, integrated, non-volatile, time-variant data store that accurate measurement requires [16]. Data quality frameworks supplement integration pipelines with completeness, accuracy, timeliness, and consistency criteria for data

integrity requirements from the data warehouse and decision support systems.

Data governance, defined as the specification of decision rights and an accountability framework for the management of data assets [17], provides the organizational structure within which data quality frameworks operate. Data timeliness is an underappreciated dimension of foundation quality. Batch processing architectures that refresh data on daily cycles are structurally incapable of supporting intra-day decision requirements [5]. The choice among batch, micro-batch, and streaming architectures therefore carries direct implications for the decision timelines the KPI system can support. This layer's success condition is not data completeness in isolation; it is the production of a data substrate whose quality and timeliness properties are explicitly matched to the decision timeline requirements of the layers above it.

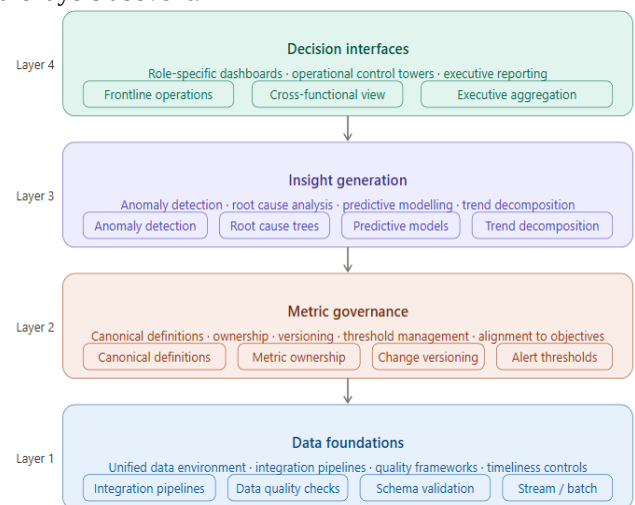


Figure 2. Layered Enterprise KPI Architecture [16, 17]

2) Metric Governance

The Metric Governance layer is responsible for defining, maintaining, and controlling the metrics that populate the KPI system. Its boundary spans the point at which raw data attributes become governed business metrics and extends to the versioned, ownership-assigned metric registry that the Insight Generation layer draws from. Its core function is to establish a canonical definition for each metric that is authoritative across all teams and systems. When governance is absent or informal, the definitional divergence described in Section III A (2) becomes structurally embedded [17].

A governance framework assigns ownership to each metric. The owner is accountable for the accuracy of the metric definition, the validity of its source data path, and the currency of its business relevance [12]. Governance documentation captures the calculation logic, source systems and fields, aggregation method, time boundaries, applicable filters, and the business objective the metric serves. Versioning is a governance requirement that is frequently omitted. When a metric definition changes, historical comparisons become invalid unless the change is logged with

a timestamp and a rationale [11]. Table III provides a structured reference for the attributes that should be captured for each governed metric.

TABLE III
METRIC GOVERNANCE ATTRIBUTE FRAMEWORK [12, 17]

Attribute	Definition	Owner Role	Review Cadence
Metric name	Canonical identifier used across all platforms	Analytics lead	Annual
Calculation logic	Formula, aggregation method, and time basis	Data engineering	On change
Source system(s)	Authoritative data origin and pipeline path	Data architecture	On change
Threshold values	Warning and critical alert boundaries	Operations lead	Quarterly
Business context	Strategic objective the metric serves	Product or process owner	Semi-annual
Versioning log	Change history with timestamp and rationale	Governance committee	Continuous

Governance also constrains metric proliferation by establishing a structured review process before new indicators are introduced [12]. The Balanced Scorecard principle that metrics should align with strategic objectives [7] remains directly applicable here: metrics without a clear line of sight to a business outcome are candidates for retirement rather than continued maintenance. This layer's success condition is not the production of governance documentation; it is the organizational adoption of canonical definitions as the authoritative source of truth across all teams that consume the metrics.

A persistent governance design challenge is the tension between standardization and unit-level flexibility. However, a strict canonical governance style risks losing definitional autonomy from business units where existing local definitions vary due to historical context, legitimate business differentiation and operational requirements. This tension is addressed in Section IV C.

3) Insight Generation

The Insight Generation layer transforms governed metrics into diagnostic and predictive outputs. Its boundary spans the point at which stable metric streams enter analytical processing and the point at which interpreted outputs, annotated with causal context and forward-looking projections, are delivered to the Decision Interfaces layer. Metrics describe what is happening; insight generation produces explanations of why performance is changing and forecasts of where it is headed [4]. This distinction is what separates a KPI architecture from a reporting system.

Statistical process control techniques establish performance baselines and identify deviations that fall outside expected variation ranges. These techniques are suited to

high-frequency operational metrics where normal variation is measurable and deviations from that baseline signal process instability [5]. Anomaly detection extends this capability to identify statistically unusual behavior without requiring the violation of a fixed threshold [3]. The three anomaly categories distinguished in the detection literature, point anomalies, contextual anomalies, and collective anomalies [18], each carry different operational implications and require different detection techniques and response protocols. Root cause analysis frameworks link observable metric deviations to their upstream operational drivers [19]. Causal inference through directed acyclic graph models and interventional reasoning [19] provides the theoretical basis for making these linkages systematically, rather than inferring causality from correlation alone.

Predictive models extend the analysis beyond the present moment. Regression models, time-series decomposition models, and machine learning classifiers are used to forecast metric trends based on historical data and leading indicators [4]. The operational value of this capability lies not in forecast precision in isolation, but in the lead time that actionable precision enables. What ultimately matters for this layer is not the sophistication of the analytical models deployed, but whether their outputs are interpretable enough to inform a specific operational decision within the time constraints that decision requires. Table IV summarizes the primary analytical techniques in this layer.

TABLE IV
ANALYTICAL TECHNIQUES IN THE INSIGHT GENERATION LAYER [4, 5, 18, 19]

Technique	Primary Function	Operational Application
Statistical process control	Detect deviations from expected performance baselines	Flag fulfillment throughput anomalies in real time
Trend decomposition	Separate seasonal, cyclical, and residual signal components	Isolate genuine capacity decline from seasonal demand shifts
Root cause analysis trees	Link symptom metrics to underlying operational drivers	Trace customer satisfaction drops to specific failure nodes
Predictive regression models	Forecast future metric values from historical patterns	Anticipate inventory shortfalls before they materialize
Anomaly detection	Automatically surface statistically unexpected behavior	Identify process drift before human review cycles catch it
Correlation mapping	Quantify inter-metric relationships and lead-lag patterns	Understand how order processing delays propagate downstream

The relationship between this layer and AI/machine learning capabilities is architecturally specific, not incidental. Anomaly detection models are embedded at this layer as consumers of governed metric streams from layer two; they require stable, canonically defined input metrics to avoid detecting governance artifacts rather than genuine operational

deviations. Predictive regression and time-series models operate on the same governed streams; definitional drift in layer two invalidates the training distribution these models were fitted to, which is why metric versioning in the governance layer is a hard dependency for sustained model performance rather than an optional documentation practice. Causal inference models, implemented through directed acyclic graphs as described by Pearl [19], require that metric definitions be stable and consistently sourced before interventional reasoning can be trusted. This dependency chain means AI and ML capabilities cannot be grafted onto a KPI system that lacks governance maturity; they are outputs of a well-functioning layer two, not replacements for it. Model bias governance, including the auditing of training data distributions, anomaly flag patterns, and root cause attribution outputs for systematic bias, is therefore an Insight Generation layer responsibility, architecturally co-located with the models that require it, rather than a Metric Governance function.

4) Decision Interfaces

The Decision Interfaces layer converts the outputs of the Insight Generation layer into formats and interaction patterns that support actual operational decisions. Its boundary spans the receipt of pre-interpreted analytical outputs and the delivery of those outputs through role-calibrated interfaces to decision-makers. This layer is the most visible to end users and is also the most frequently designed without reference to the decision workflows it is intended to serve [2].

An effective decision interface is not a comprehensive data display. It is a focused presentation of the signals most relevant to a specific decision type, providing sufficient context to interpret those signals and identify the appropriate response [3]. Dashboard design research establishes that visual complexity beyond the decision-relevant threshold actively degrades decision quality [2]. The interface layer must therefore receive filtered, pre-interpreted outputs from the Insight Generation layer rather than raw metric streams. The relationship between these two layers is directional but not unidirectional: decision outcomes and user override actions from the interface layer should feed back to the governance and baseline recalibration functions in the layers below.

Role specificity is a foundational design principle. A frontline logistics manager making intra-day routing decisions requires different information at a different level of granularity than an operations director conducting a weekly performance review [10]. Effective KPI architectures maintain multiple interface configurations, each calibrated to the decision timelines, information requirements, and authority levels of a specific operational role. The operational control tower is one application of this principle: a multi-dimensional decision interface that integrates cross-functional metrics into a shared operational view [13]. Strategic reporting requires a different interface configuration from operational decision support. A properly designed KPI architecture computes aggregations from the same base data

as the operational dashboards, eliminating the reconciliation problems that arise when strategic and operational reporting draw from separate sources. Figure 3 depicts the escalation framework governing decision authority and signal routing across interface tiers.

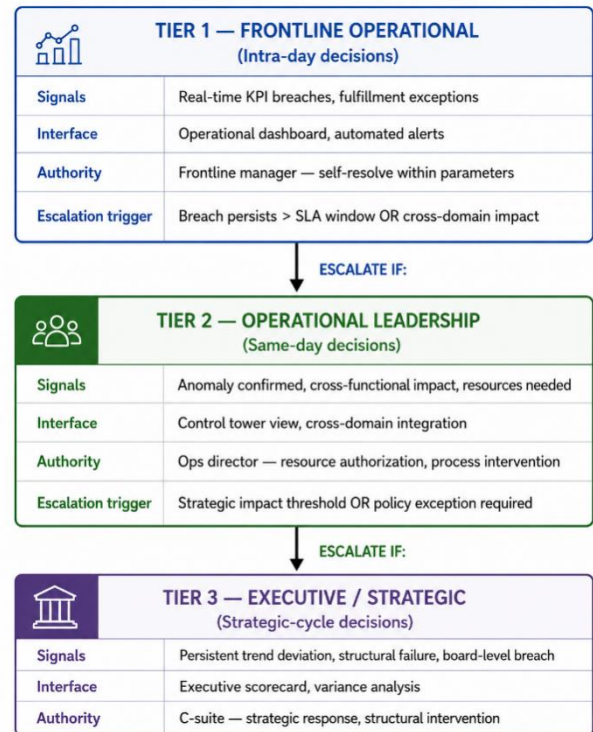


Figure 3. Operational Escalation Framework: Decision Authority and Signal Routing Across KPI Architecture [7, 10, 14]

C. Design Principles

The four-layer structure can be distilled into four design principles that summarize the architectural logic governing layer sequencing and dependency. These principles are offered as design guidance rather than as a strict implementation mandate, and they are intended to function as a compact decision rule set for architects and program sponsors evaluating implementation sequencing.

1) *Govern before measuring.* Metrics computed without a canonical definition layer in place produce numbers that cannot be trusted across teams, regardless of analytical sophistication applied afterward.

2) *Standardize before optimizing.* Analytical optimization applied to inconsistently defined metrics amplifies definitional disagreement rather than resolving it; standardization is a precondition for, not a byproduct of, analytical investment.

3) *Diagnose before prescribing.* Prescriptive recommendations built on top of descriptive or undiagnosed signals inherit the absence of causal grounding, and inherit the risk of recommending interventions against the wrong root cause.

4) *Deliver decisions rather than dashboards.* The architecture's success condition is a role calibrated decision interface that supports a specific action, not a comprehensive visualization surface that defers interpretation to the viewer.

D. Illustrative Implementations Across Industry Contexts

The implementation illustrations in this section are synthesized composite scenarios derived from patterns documented repeatedly across the cited literature, rather than from single empirical deployments. Two industry contexts are used to demonstrate the generalizability of the architectural model: logistics and distribution, and discrete manufacturing. The operational KPI ranges and numerical values presented throughout are illustrative approximations, drawn from recurring patterns in the literature, intended to demonstrate architectural impact pathways rather than to represent statistically validated or measured outcomes from a specific organization.

1) Logistics and Distribution Context

A mid-sized regional logistics and distribution company relied on functionally siloed dashboards for performance measurement, with separate metrics maintained across operations, finance, and customer service. Cross-functional metric reconciliation depended on a manual weekly process, and operational data was refreshed on 24-hour batch cycles. This structural pattern is identified by Davenport as a leading cause of low analytics adoption in organizations with otherwise capable data infrastructure [1].

TABLE V
PRE- AND POST-DEPLOYMENT KPI OUTCOMES: LOGISTICS AND DISTRIBUTION CONTEXT (COMPOSITE ILLUSTRATION) [1, 5, 6, 10, 11, 12, 17, 18]

KPI	Pre-Deployment Baseline	Post-Deployment Outcome
Decision latency (anomaly response)	28 to 34 hours	Under 3 hours
Cross-functional review prep time	Approx. 4 hours per cycle	Under 75 minutes
Fulfillment exception escalation rate	Baseline index 100	Reduced approx. 34%
Metric reconciliation disputes per review	6 to 9 per session	Under 1 per session
Anomaly detection mode	24-hour batch cycle	Real-time threshold alert
Dashboard estate actively used	Approx. 40% of indicators	Approx. 85% of indicators

Deployment proceeded in four sequential phases corresponding to the layer order described in Section III B: migration from batch ETL to streaming data integration; establishment of a canonical metric governance registry with defined ownership and versioning covering 27 key business indicators; development of statistical baselines for fulfillment metric anomaly detection; and deployment of role-based decision interfaces replacing generic dashboards. This sequencing reflects the finding that phased implementation

based on operational need is associated with sustained KPI program value [11].

The outcomes reflected patterns documented in the analytics maturity literature. Governance maturity is associated with faster decision cycles and reduced cross-functional reconciliation burden [6, 11]. Canonical definitions eliminated inter-team metric interpretation disagreements during performance reviews. Automated alerting (threshold-based) replaced the 24-hour manual review cycle to respond to detected anomalies, consistent with the detection and response patterns of Chandola et al. [18]. The baseline and post-deployment KPI ranges are seen in Table V.

2) Discrete Manufacturing Context

A mid-market discrete manufacturing operation producing engineered components for industrial customers, maintained production performance metrics across eight shop floor cells, with separate reporting systems for quality, throughput, and maintenance. Metrics were owned by functional area supervisors and were not governed by a shared definitional registry. Overall equipment effectiveness was calculated using three distinct methods across production cells, making cross-cell comparison unreliable and preventing aggregate reporting that leadership could act on with confidence.

TABLE VI
PRE- AND POST-DEPLOYMENT KPI OUTCOMES: DISCRETE MANUFACTURING CONTEXT (COMPOSITE ILLUSTRATION) [5, 6, 9, 11, 12, 17, 18]

KPI	Pre-Deployment Baseline	Post-Deployment Outcome
OEE cross-cell comparability	Three incompatible methods	Single governed definition, fully comparable
Quality failure lead time to detection	End-of-shift batch review	Real-time threshold alert within cycle
Root cause identification time	4 to 6 hours per incident	Under 90 minutes with causal tree
Cross-functional review disputes (OEE)	4 to 7 per session	Under 1 per session
Supervisors actively using role dashboards	Approx. 35%	Approx. 80%

The deployment followed the same four-phase sequence. Data foundations work focused on integrating SCADA system outputs, quality management system records, and ERP production order data into a unified analytical store with defined latency targets by metric class. Governance work focused on a single canonical definition for overall equipment effectiveness, agreed upon across all eight cells, and on establishing threshold values for availability, performance rate, and quality rate as separate governed sub-metrics with independent ownership. The insight generation layer introduced anomaly detection on cycle time deviation and first-pass yield, with root cause trees linking yield failures to upstream material quality and tooling state indicators. Decision interfaces were configured separately for cell-level

shift supervisors, who required real-time single-cell views, and for production managers, who required cross-cell comparative views on a 4-hour rolling basis.

The outcomes in this context reflected a different benefit profile than the logistics deployment. The primary gains were in cross-cell comparability and in the lead time available before quality failures propagated to shipment-level impacts. Governance adoption in this context required more active change management because cell supervisors had historically treated their local definitions as operationally accurate, and the harmonization process required demonstrating that the canonical definition preserved rather than discarded that local accuracy. This pattern is consistent with Bourne et al.'s finding that the political dimension of metric ownership must be treated as a central element of the governance adoption process rather than a side concern [11].

IV. DISCUSSION

A. Implementation Conditions and Organizational Prerequisites

Enterprise KPI Architecture is not implemented once and maintained passively. It is a living system that requires sustained investment across technical, organizational, and governance dimensions [11]. The most common technical barrier is the existence of organizational data silos. Resolving them requires both technical and organizational interventions. Integration platforms, API standardization, and event streaming architectures establish the data movement pathways needed for a unified metric environment [16]. Data sharing agreements and cross-functional governance bodies are prerequisites for success [17]. The specification of decision rights over data assets is the foundational act of data governance [17], and where data ownership lacks clarity, integration initiatives tend to remain in the domain of organizational negotiation rather than technical execution.

Legacy systems compound this challenge. Core operational platforms were designed before modern data integration models existed. Many metric data sources lack standard APIs, publish data in proprietary formats, or operate on periodic batch cycles [16]. The most common architectural response involves an abstraction layer that normalizes legacy source outputs within the shared data environment without requiring modification of source systems.

Scalability in cloud native environments is a related design consideration that is increasingly central to the Data Foundations and Insight Generation layers. Elastic compute and storage allow the streaming and micro batch pathways described above to scale with operational data volume without the capacity planning overhead required by fixed on premises infrastructure. Many enterprises operate across more than one cloud provider, whether through deliberate multi cloud strategy or through accumulated acquisitions, and governing metric pipelines consistently across providers such as AWS, Azure, and GCP requires that the Metric Governance layer's canonical definitions and ownership assignments be enforced independently of the underlying

compute environment in which a given pipeline happens to run. Data lineage tracking, the ability to trace a governed metric back through every transformation to its originating source field, and metadata management, the cataloging of schema, ownership, and refresh cadence for each data asset, are governance functions that become more difficult and more necessary as the number of source systems and cloud environments grows. Knowledge graphs are an emerging mechanism for representing the relationships among governed metrics, their upstream data sources, and their downstream consumers, and they offer a structured alternative to maintaining those relationships as informal documentation; a metric relationship graph can support both root cause tracing in the Insight Generation layer and impact analysis when a metric definition changes.

Metric governance frameworks fail most often on organizational rather than technical grounds. Governance adoption requires visible leadership sponsorship and a change management process that involves impacted teams in framework design [12]. This is consistent with broader findings in organizational change management: participation in design increases commitment to implementation [14]. Training must be treated as a continuous investment rather than a one-time activity [5]. A further implementation failure mode specific to KPI architecture programs is the deferral of operational value while architectural completeness is pursued [13]. A more durable approach sequences implementation by operational priority, releasing a functional decision interface across a narrow scope before extending the architecture further. Figure 4 depicts the recommended phased implementation roadmap.

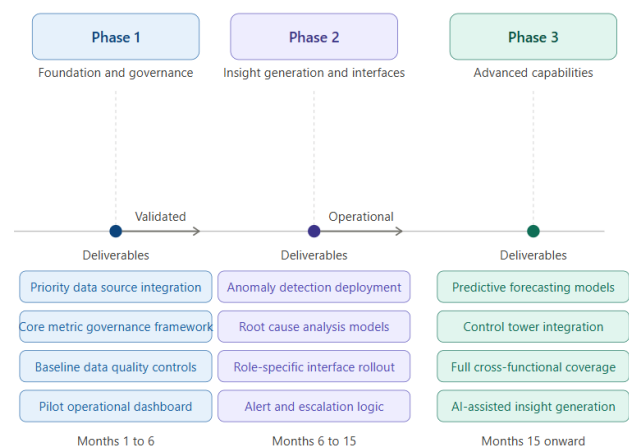


Figure 4. Phased Implementation Roadmap [11, 12, 13]

Organizations at early stages of data maturity face a sequencing challenge that the phased roadmap partially addresses but does not fully resolve. The Metric Governance layer cannot produce consistent outputs if the Data Foundations layer does not yet deliver stable, timeliness-matched source data; attempting to deploy governance before foundational data integration is mature results in governing metrics that are structurally unreliable. For these

organizations, the recommended entry point is a bounded Data Foundations investment limited to the two or three highest-priority metric domains, establishing a minimum viable substrate before governance work begins. Implementation risks specific to this architecture extend beyond technical barriers. Organizational resistance to canonical definitions is common when business units have historically treated local metric variants as operationally authoritative. Conflicting KPI definitions accumulated over years of independent measurement practice carry political as well as technical dimensions, as Bourne et al. document [11]. System integration costs for legacy operational platforms without standard APIs can exceed initial estimates significantly; the abstraction layer approach described above reduces but does not eliminate this exposure. These risks are not arguments against implementation; they are arguments for sequencing, scoping, and stakeholder engagement strategies that are realistic about the timeline from architectural investment to decision-support value.

Each layer has identifiable implementation success indicators that can be monitored without waiting for full-architecture deployment. Data Foundations success is indicated by data completeness rates, source-to-substrate latency against defined targets, and schema drift incident frequency. Metric Governance success is indicated by the proportion of active metrics with assigned ownership, version log currency, and the rate of cross-functional review sessions that conclude without metric interpretation disputes. Insight Generation success is indicated by anomaly detection precision and recall against known operational events, root cause identification time relative to pre-deployment baselines, and the proportion of model outputs rated interpretable by operational users. Decision Interfaces success is indicated by active dashboard utilization rates, decision latency from signal to operational response, and escalation routing accuracy. These indicators allow program sponsors to evaluate architectural progress layer by layer, rather than deferring all measurement to a final deployment state.

B. Organizational Impact and Emerging Analytical Capabilities

The literature on analytics maturity programs reports improvements across several operational dimensions in organizations that implement structured KPI governance [6]. The most consistently documented pattern is a reduction in the time required to formulate a response to performance deviations, attributable to pre-interpreted signals and threshold-triggered alerts that surface decision-relevant information without requiring manual navigation of raw metric streams [1]. Cross-functional alignment improvement is documented by Bourne et al. as a direct outcome of shared metric definitions reducing interpretation disputes [11]. Leadership trust in analytics systems is consistent with the organizational patterns Davenport identifies in analytics-competing organizations [1], where definitional consistency and data quality underpin the credibility of analytical outputs.

These patterns from the literature are the basis for the directional claims in this article's illustrative implementations; they are not independent empirical findings generated by this study. The composite deployment scenarios in Section III D are designed to show that the architectural mechanisms proposed here are consistent with the patterns those studies document, not to constitute additional empirical evidence.

The integration of machine learning and AI capabilities into KPI architectures is accelerating. Anomaly detection models embedded at the Insight Generation layer already represent a feature of mature systems [18]. Hybrid intelligence frameworks represent the near-term trajectory for advanced KPI implementations, combining machine-generated analytical outputs with human judgment in structured decision workflows [15]. In a hybrid intelligence system, machine capabilities handle data-intensive analytical tasks while human judgment addresses context-dependent interpretation [15]. Prescriptive analytics depends on causal modeling capability to recommend specific operational interventions based on the predicted consequences of available options [19].

Looking two to three years ahead, the analytical capabilities embedded at the Insight Generation and Decision Interfaces layers are likely to extend beyond anomaly detection and hybrid intelligence toward a generation of language model-based capabilities. Large Language Models [21] offer a natural interface for translating governed metric outputs into narrative explanations that operational decision makers can interrogate conversationally rather than only through structured dashboards. Retrieval Augmented Generation [22] extends this capability by grounding those explanations in the governed metric registry, the versioning log, and the documented business context for each metric, reducing the risk that a generated explanation drifts from the canonical definition it is meant to describe. AI agents capable of multi-step reasoning could traverse the data to decision pathway autonomously, querying governed metrics, retrieving relevant diagnostic context, and assembling a recommended response for human review within the decision interface layer rather than requiring an analyst to assemble that context manually. This trajectory points toward agentic decision support, in which the architecture's governance and decision rights structures become the control surface that determines how much autonomy an AI agent may exercise before a human decision maker must intervene. The governance principles established in this article, particularly the explainability, audit logging, and human override requirements discussed in the Limitations section, are positioned as a necessary foundation for that future state rather than as a constraint that these emerging technologies will eventually outgrow.

The governance control surface described above should extend to explicit staging of AI autonomy across decision types, rather than applying a uniform oversight requirement to every automated output. A mature governance model calibrates the degree of autonomy an agent may exercise to

the reversibility and consequence severity of the decision at hand: a low consequence, easily reversible action, such as flagging an anomaly for review, can tolerate greater autonomy than a high consequence, difficult to reverse action, such as rerouting capacity away from a committed customer order. This staged approach is consistent with emerging external frameworks for AI risk management, including the NIST AI Risk Management Framework [23] and the EU AI Act's tiered risk classification [24], both of which condition the degree of required oversight on the potential impact of an erroneous output. Embedding this staging logic within the Metric Governance and Decision Interfaces layers gives organizations a structured basis for deciding, in advance, which categories of AI recommendation require mandatory human sign off, which require post hoc audit, and which may proceed without individual review subject to aggregate monitoring. Absent this structure, organizations tend to apply oversight inconsistently, either slowing every AI supported decision with unnecessary review or allowing high stakes recommendations to proceed without adequate scrutiny.

Organizational decision-making structures must evolve alongside these analytical capabilities [14]. The boundaries between automated decision execution and human oversight must be explicitly designed into the system during the architecture phase, not added retrospectively [14]. Real-time streaming architectures will increasingly displace batch-oriented data foundations as the operational baseline [5], enabling the intra-day decision support that conventional architectures structurally cannot provide. Figure 5 depicts the AI-assisted KPI decision loop, from signal detection to governed response.

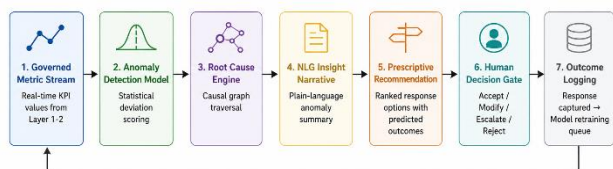


Figure 5: AI-Assisted KPI Decision Loop: From Signal Detection to Governed Response [13, 14, 15, 18, 19]

C. Governance Standardization and Business Unit Flexibility

Strict canonical governance and business unit operational flexibility represent a genuine trade-off, not merely a change management challenge, because local metric definitions in many organizations encode operationally meaningful distinctions that a uniform definition may not preserve.

A persistent design tension in metric governance is the conflict between canonical standardization and the legitimate need for operational flexibility across business units. Strict governance enforces uniform definitions, which is necessary for cross-functional comparability and for the integrity of aggregated strategic reporting. However, business units that have historically maintained their own metric definitions may have embedded operationally meaningful distinctions that a purely standardized definition cannot preserve. Treating this

tension as simply a change management problem underestimates its technical and analytical dimensions.

A workable resolution distinguishes between enterprise-governed core metrics and unit-governed supplementary metrics. Core metrics are those used in cross-functional comparisons, executive reporting, or regulatory compliance: they require canonical definitions and centralized governance. Supplementary metrics are those used exclusively within a single business unit for local operational management: they can carry unit-defined definitions without undermining cross-functional integrity, provided they are clearly classified as local and excluded from enterprise aggregations. The governance framework must specify which metrics belong in each category and must include a review process for reclassification as business needs evolve.

The manufacturing deployment scenario in Section III D (2) illustrates this distinction in practice. Cell-level supervisors retained the ability to define local production sub-metrics for shift management purposes, while the canonical overall equipment effectiveness definition was enforced only at the cell-level rollup used in cross-cell comparisons. This two-tier approach preserved local operational utility while establishing the comparability that management reporting required. Bourne et al. identify the politics of metric ownership as a central rather than peripheral element of governance adoption [11]; the two-tier design reduces the ownership conflict by limiting canonical governance to the metrics where cross-functional consistency is genuinely required.

D. Positioning Relative to Prior Research

The framework proposed in this article extends several established research threads. The Balanced Scorecard [7] and the performance measurement design literature [8, 9, 11] established that strategic alignment and structured governance are prerequisites for effective measurement systems. This article builds on those foundations by addressing what must follow once governance is in place: how metrics must be connected to diagnostic and predictive analytical layers to produce decision support rather than reporting. The Balanced Scorecard was designed as a strategic management system operating on quarterly or annual cycles and does not address the integration architecture, data latency requirements, or analytical methods required for intra-day operational decision-making. This article extends the governance alignment principle to the operational level by embedding it within a streaming data foundation and applying diagnostic analytics to governed metric streams.

The Business Intelligence architecture literature [16] establishes the data warehouse as a core component for integrated operational measurement, and the analytics capability literature [6] describes the organizational preconditions for realizing analytics value. This article extends that tradition by treating the decision interface layer as an architectural layer in its own right, with explicit role-calibration requirements. Where existing BI architectures treat dashboards as visualization products, this framework

treats decision interfaces as role-calibrated decision support systems with defined requirements governing signal filtering, alert hierarchy, and escalation pathway design [2, 10].

Enterprise Performance Management frameworks provide financial planning and consolidation architecture but do not address the operational decision support requirements, streaming data integration, or anomaly detection capabilities that are central to this framework. Decision intelligence theory [13] provides a formal model for connecting analytical output to structured decision outcomes. This article operationalizes that model within a KPI stack architecture by specifying where in the stack decision intelligence capabilities should be deployed, the governance conditions they must satisfy, and the decision oversight boundaries required to maintain human accountability in AI-supported decision processes [14, 15]. The contribution of this article is the integration of these streams into a unified, layered architectural model with explicit design requirements at each layer and a documented set of implementation conditions.

E. Limitations

The Enterprise KPI Architecture framework carries several limitations that must be acknowledged explicitly.

First, and most importantly, the Enterprise KPI Architecture framework is a conceptual contribution developed through design-science research methodology. It has not been empirically validated through controlled deployment in real organizational settings, and the conclusions drawn in this article are bounded to what the literature synthesis and internal consistency analysis support. The composite cases in Section III D are architectural illustrations, not empirical deployments of this framework. In a similar way, the KPI ranges are illustrative estimates meant to capture the ranges of operational performance found in the literature, rather than actual deployments. Future studies should prioritize empirical case validation across industry sectors, with pre- and post-deployment measurement of the specific outcomes the framework claims to address.

Second, the framework assumes organizational conditions that are neither universal nor inexpensive to establish. Computing metrics in near real time across enterprise-scale source systems requires significant infrastructure investment. In organizations where elastic scaling is not warranted, the engineering complexity of distributed stream processing relative to batch processing may not be justified [5]. A tiered latency model, in which high-criticality signals are streamed and lower-criticality signals are aggregated in micro-batches, partially addresses this constraint but does not eliminate it.

Third, KPI architectures that embed machine learning-based anomaly detection and prescriptive recommendation capabilities must address explainability and accountability risks through their governance frameworks. Recommendations generated by opaque models are not traceable to causal reasoning. When such recommendations inform decisions that produce adverse outcomes, the accountability of the operational decision cannot be

established, and compliance with regulatory obligations in regulated environments cannot be assured [14]. Model governance frameworks should specify explainability standards, audit logging requirements, and human override mechanisms as first-class governance properties rather than afterthoughts. The tendency of human operators to defer to algorithmic outputs without applying independent judgment, identified as automation bias by Shrestha et al. [14], is a structural risk in human-AI decision systems that governance design must proactively address.

Fourth, the framework does not specify the quantitative thresholds of data governance maturity at which each layer can be reliably implemented. Organizations at early stages of governance maturity may be unable to sustain the governance adoption required for the Metric Governance layer before attempting the Insight Generation layer. Future research should investigate whether a governance maturity staging model, specifying the minimum capability threshold at each phase boundary, can reduce implementation failure risk for organizations at varying maturity levels.

Fifth, realizing the full value of the four-layer architecture requires sustained investment over multiple years. Each successive implementation phase incurs retraining, change management, and legacy system support costs that can erode organizational momentum. Phased delivery partially mitigates this, but the mitigation is incomplete. Program leadership must resist pressure to demonstrate the full capability before its foundational conditions have been established [12], and organizational sponsors must maintain commitment through the phases that precede visible decision-support benefits.

Sixth, the framework links KPIs to predictive and diagnostic analytical capabilities but does not address the risk that model bias in the underlying analytical models could systematically distort the outputs surfaced at the Insight Generation and Decision Interfaces layers. A model trained on historically biased operational data will produce biased anomaly flags and biased root cause attributions. These errors will be delivered through the role-calibrated interfaces of layer four with the same authority as unbiased outputs, unless the governance framework includes explicit model bias auditing requirements. Future implementations should specify bias monitoring as a first-class governance function within the Insight Generation layer, where analytical models are trained, evaluated, and deployed, rather than as a retrospective quality check on model outputs. Metric lineage and versioning remain Metric Governance responsibilities, providing the stable, traceable input definitions on which unbiased model training depends.

V. CONCLUSION

The challenge facing performance measurement systems is not data scarcity; it is architectural insufficiency. Traditional KPI approaches were designed for retrospective reporting and remain structurally ill-suited to the speed and complexity of modern operational environments. Enterprise KPI Architecture addresses this gap by organizing data

foundations, metric governance, insight generation, and decision interfaces into a coherent, purpose-built system. Each layer resolves a specific class of structural failure that conventional reporting cannot address, and together they shift the function of performance measurement from documenting outcomes to enabling decisions. As detailed in Section IV D, this framework extends the Balanced Scorecard, Business Intelligence architecture, and decision intelligence theory into a single layered model with explicit human accountability boundaries. The two-tier governance model for standardization versus unit flexibility, and the explicit treatment of AI explainability and automation bias risks as governance design requirements, represent contributions beyond the existing literature in each of these traditions. The implementation sequencing and organizational conditions identified here extend documented findings on analytics deployment maturity by specifying the architectural design choices that determine whether KPI programs sustain value beyond initial deployment. Future studies should focus on empirical case validation across industry sectors, on quantifying the governance maturity thresholds at which each architectural layer can be reliably implemented, and on measuring the organizational capability requirements for absorbing AI-assisted prescriptive analytics at scale. Enterprise KPI Architecture represents a shift from measuring organizational performance to engineering organizational decision capability, establishing a foundation upon which future AI enabled operational systems can be built.

REFERENCES

- [1] T. H. Davenport, "Competing on Analytics," *Harvard Business Review*, 2006. Available: <https://cs.brown.edu/courses/cs295-11/competing.pdf>
- [2] S. A. Melnyk, U. Bititci, K. Platts, J. Tobias, & B. Andersen, "Is performance measurement and management fit for the future?" *Management Accounting Research*, vol. 25, no. 2, pp. 173–186, 2014. Available: <http://dx.doi.org/10.1016/j.mar.2013.07.007>
- [3] H. Chen, R. H. L. Chiang, & V. C. Storey, "Business intelligence and analytics: From big data to big impact," *MIS Quarterly*, vol. 36, no. 4, pp. 1165–1188, 2012. Available: <https://doi.org/10.2307/41703503>
- [4] F. Provost & T. Fawcett, *Data Science for Business: What You Need to Know About Data Mining and Data-Analytic Thinking*. Sebastopol, CA: O'Reilly Media, 2013. Available: <https://fosterprovost.com/wp-content/uploads/2019/06/Data-Science-for-Business.pdf>
- [5] R. Sharda, D. Delen, & E. Turban, *Analytics, Data Science, & Artificial Intelligence: Systems for Decision Support*. London, UK: Pearson, 2021. Available: <https://www.pearson.com/content/dam/one-dot-com/one-dot-com/us/en/files/Global/Pearson-2022-IT-International-Catalogue.pdf>
- [6] B. Wixom et al., "The current state of business intelligence in academia: The arrival of big data," *Communications of the Association for Information Systems*, vol. 34, no. 1, 2014. Available: <https://doi.org/10.17705/1CAIS.03401>
- [7] R. S. Kaplan & D. P. Norton, "The balanced scorecard: Measures that drive performance," *Harvard Business Review*, 2005. Available: <https://hbr.org/2005/07/the-balanced-scorecard-measures-that-drive-performance>
- [8] A. Neely, "The performance measurement revolution: Why now and what next?" *International Journal of Operations and Production Management*, 1999. Available: <https://doi.org/10.1108/01443579910247437>
- [9] M. S. Krishnan & M. I. Kellner, "Measuring process consistency: Implications for reducing software defects," *IEEE Transactions on Software Engineering*, 2002. Available: <https://ieeexplore.ieee.org/document/824401>
- [10] E. Brynjolfsson & K. McElheran, "The rapid adoption of data-driven decision-making," *American Economic Review*, vol. 106, no. 5, pp. 133–139, 2016. Available: <http://dx.doi.org/10.1257/aer.p20161016>
- [11] M. Bourne, J. Mills, M. Wilcox, A. Neely, & K. Platts, "Designing, implementing and updating performance measurement systems," *International Journal of Operations & Production Management*, vol. 20, no. 7, pp. 754–771, 2000. Available: <https://doi.org/10.1108/01443570010330739>
- [12] D. Parmenter, *Key Performance Indicators: Developing, Implementing, and Using Winning KPIs*. Hoboken, NJ: John Wiley & Sons, 2015. Available: https://kpiacademy.uk/wp-content/uploads/2022/04/Key-Performance-Indicators-KPI-Developing-Implementing-and-Using-Winning-KPIs-David-Parmenter-z-lib.org_.pdf
- [13] L. Pratt, C. Bisson, & T. Warin, "Bringing advanced technology to strategic decision-making: The Decision Intelligence/Data Science (DI/DS) integration framework," *Futures*, vol. 152, 2023. Available: <https://www.sciencedirect.com/science/article/abs/pii/S0016328723001222>
- [14] Y. R. Shrestha, S. M. Ben-Menahem, & G. Von Krogh, "Organizational decision-making structures in the age of artificial intelligence," *California Management Review*, vol. 61, no. 4, pp. 66–83, 2019. Available: <https://doi.org/10.1177/0008125619862257>
- [15] D. Dellermann, P. Ebel, M. Söllner, & J. M. Leimeister, "Hybrid intelligence," *Business & Information Systems Engineering*, vol. 61, no. 5, pp. 637–643, 2019. Available: <https://arxiv.org/pdf/2105.00691>
- [16] W. H. Inmon, *Building the Data Warehouse*. Hoboken, NJ: John Wiley & Sons, 2005. Available: http://www.r-5.org/files/books/computers/databases/warehouses/W_H_Inmon-Building_the_Data_Warehouse-EN.pdf
- [17] V. Khatri & C. V. Brown, "Designing data governance," *Communications of the ACM*, 2010. Available: <https://dl.acm.org/doi/pdf/10.1145/1629175.1629210>
- [18] V. Chandola, A. Banerjee, & V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys*, 2009. Available: <https://dl.acm.org/doi/pdf/10.1145/1541880.1541882>
- [19] J. Pearl, "Causality: Models, reasoning, and inference," *Econometric Theory*, 2003. Available: <https://doi.org/10.1017/S0266466603004109>
- [20] A. R. Hevner, S. T. March, J. Park, & S. Ram, "Design science in information systems research," *MIS Quarterly*, vol. 28, no. 1, 2004. Available: <https://doi.org/10.2307/25148625>
- [21] T. B. Brown et al., "Language Models are Few-Shot Learners," *Advances in Neural Information Processing Systems*, vol. 33, pp. 1877–1901, 2020. Available: <https://arxiv.org/abs/2005.14165>
- [22] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020. Available: <https://arxiv.org/abs/2005.11401>
- [23] National Institute of Standards and Technology, "Artificial Intelligence Risk Management Framework (AI RMF 1.0)," NIST, 2023. Available: <https://doi.org/10.6028/NIST.AI.100-1>
- [24] European Parliament and Council, "Regulation (EU) 2024/1689 of the European Parliament and of the Council Laying Down Harmonised Rules on Artificial Intelligence (Artificial Intelligence Act)," *Official Journal of the European Union*, 2024. Available: <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32024R1689>