



International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 14, Issue 11, November 2025

Predictive Landscape Intelligence: Leveraging Machine Learning for Proactive Anomaly Detection, Capacity Forecasting, and Incident Prevention in Multi-Component Enterprise ERP Environments

Srinivas Kakarla

SAP Architect, USA

ABSTRACT: Multi-component enterprise resource planning (ERP) environments - spanning core transactional modules, middleware, database clusters, identity services, and API gateways - generate telemetry at a volume and complexity that traditional threshold-based monitoring cannot fully exploit. This article presents a machine learning-based approach to landscape intelligence organized around three capabilities: proactive anomaly detection using supervised and unsupervised models trained on component telemetry, capacity forecasting using time-series methods with explicit uncertainty bands, and incident prevention through automated and assisted response integrated with existing ERP operations tooling. We present a reference ML pipeline architecture, detailed model evaluation artifacts (ROC curves, confusion matrices, precision/recall trade-off curves), an illustrative component health grid across a twelve-component landscape, and a five-phase adoption roadmap. All quantitative model-performance and before/after figures are presented as illustrative estimates for methodological demonstration, clearly flagged throughout, rather than as measured results from a specific production deployment.

KEYWORDS: Enterprise Resource Planning, Machine Learning, Anomaly Detection, Capacity Forecasting, Predictive Maintenance, AIOps, Observability, Time-Series Forecasting, Incident Prevention, Landscape Intelligence

I. INTRODUCTION: BEYOND THRESHOLD-BASED MONITORING

Traditional ERP monitoring relies heavily on static thresholds: alert when CPU utilization exceeds 85%, alert when a batch job runs longer than its historical maximum, alert when disk usage crosses a fixed watermark. This approach is simple to implement and reason about, but it has two structural limitations that become more acute as ERP landscapes grow in component count and interdependency. First, static thresholds cannot distinguish a genuinely anomalous condition from a predictable seasonal pattern - a batch scheduler running near capacity every month-end close is expected, not anomalous, yet a fixed threshold treats both identically. Second, thresholds are inherently reactive: by the time a metric crosses its alert boundary, the underlying condition has often already begun to affect business processes. Machine learning-based landscape intelligence addresses both limitations by learning the normal behavior pattern of each monitored component - including its seasonal and cyclical structure - and by forecasting forward from current trends rather than only reacting to threshold breaches. This article presents an architecture and methodology for this approach, organized around three capabilities: anomaly detection (Sections 4 and 6), capacity forecasting (Section 5), and incident prevention through automated or assisted response (Section 7).

The remainder of this article proceeds as follows. Section 2 frames the specific observability challenges of multi-component ERP landscapes. Section 3 presents a reference ML pipeline architecture. Section 4 details anomaly detection methods and evaluation artifacts. Section 5 covers capacity forecasting with uncertainty quantification. Section 6 discusses feature engineering. Section 7 addresses the transition from detection to automated action. Section 8 examines component-level behavior patterns. Section 9 presents illustrative outcomes. Section 10 offers an adoption roadmap. Section 11 discusses risks and open questions, and Section 12 concludes.

II. THE MULTI-COMPONENT ERP OBSERVABILITY CHALLENGE

A modern ERP landscape rarely consists of a single monolithic system. Figure 1 presents an illustrative component health grid spanning twelve components typical of a large ERP landscape - core transactional modules (finance, logistics, HR), middleware and integration infrastructure, database and application server tiers, batch scheduling, reporting, identity and access management, API gateways, storage services, and network edge components.

Figure 1. Illustrative Component Health Grid Across a Multi-Component ERP Landscape



Figure.1. Illustrative component health grid across a twelve-component ERP landscape, color-coded by an illustrative composite health score

2.1 Why Component Interdependency Complicates Monitoring

As Figure 1 illustrates, component health is rarely uniform across a landscape - some components (identity and access management, ERP core HR) may show consistently strong health scores while others (batch/job scheduler, middleware/integration bus) show more volatile or degraded scores. Because components are interdependent, a degradation in one - particularly a shared, upstream component like a database cluster or middleware bus - can cascade into symptoms observed in several downstream components, complicating root-cause attribution when relying on component-level monitoring alone.

2.2 The Case for Cross-Component Feature Engineering

This interdependency is a primary motivation for the feature engineering approach detailed in Section 6: rather than modeling each component in isolation, a landscape intelligence system should incorporate features that capture cross-component relationships - for example, the correlation between database lock wait time and application-tier response

latency - so that anomaly detection models can distinguish a component-local issue from a cascading, landscape-wide one.

Table.1. Comparison of Observability Approaches Across Key Capabilities.

Observability Approach	Seasonal Detects Patterns?	Cross- Component Detects Cascades?	Reactive or Proactive?
Static threshold alerting	No	No	Reactive
Statistical baselining (moving average)	Partially	No	Reactive
Supervised anomaly detection (this article)	Yes	Yes, with cross- component features	Proactive
Time-series forecasting (this article)	Yes	Limited without multivariate inputs	Proactive

III. A REFERENCE ML PIPELINE FOR LANDSCAPE INTELLIGENCE

Figure 2 presents a five-stage reference pipeline: data ingestion from logs, metrics, traces, and events; feature engineering including windowing and normalization; model inference for both anomaly detection and forecasting; alert scoring and triage; and automated or assisted response. A feedback loop returns labeled outcomes - whether an alert was a true positive requiring action or a false positive that should inform future scoring - back into feature engineering and model retraining.

Figure 2. Reference ML Pipeline for Landscape Intelligence, with a Retraining Feedback Loop

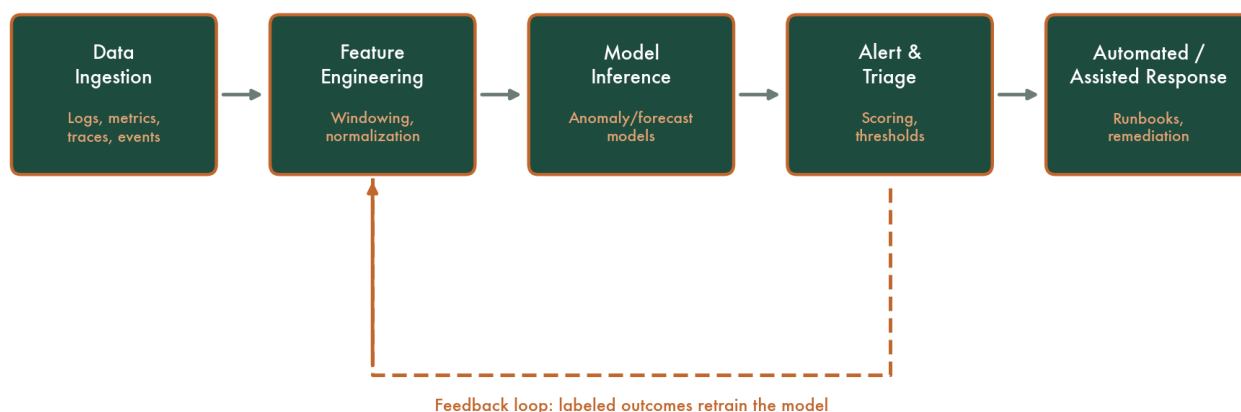


Figure.2. Reference ML pipeline for landscape intelligence, showing the five processing stages and the retraining feedback loop from alert outcomes back to feature engineering

3.1 Why the Feedback Loop Is Not Optional

A landscape intelligence system deployed without a structured feedback mechanism will degrade in perceived usefulness over time even if the underlying model does not change, because the operational environment it monitors will drift - new components are added, workload patterns shift with business growth, and one-time events (a migration,

a reorganization) can permanently alter what "normal" behavior looks like. The feedback loop shown in Figure 2 is the mechanism by which labeled outcomes (confirmed true positives, confirmed false positives, missed anomalies later identified by operators) flow back into the training data, allowing the model to be periodically retrained on data that reflects current landscape behavior rather than the conditions present at initial training time.

MODEL CARD - Landscape Anomaly Detector v1

MODEL CARD - Landscape Anomaly Detector v1	
Model family	Gradient-boosted ensemble (illustrative)
Training data window	Rolling 90 days of component telemetry (illustrative)
Retraining cadence	Weekly, triggered by feedback-loop labels (illustrative)
Primary inputs	12 engineered features across component and cross-component signals
Illustrative AUC	0.94 (see Figure 6)
Human-in-the-loop	Alerts above a configurable threshold route to on-call review

IV. ANOMALY DETECTION: METHODS AND EVALUATION

Figure 3 illustrates anomaly detection applied to a single component's telemetry over one week, showing an expected range band derived from historical seasonal patterns and six flagged anomalies where the observed metric departed materially from that expected range.

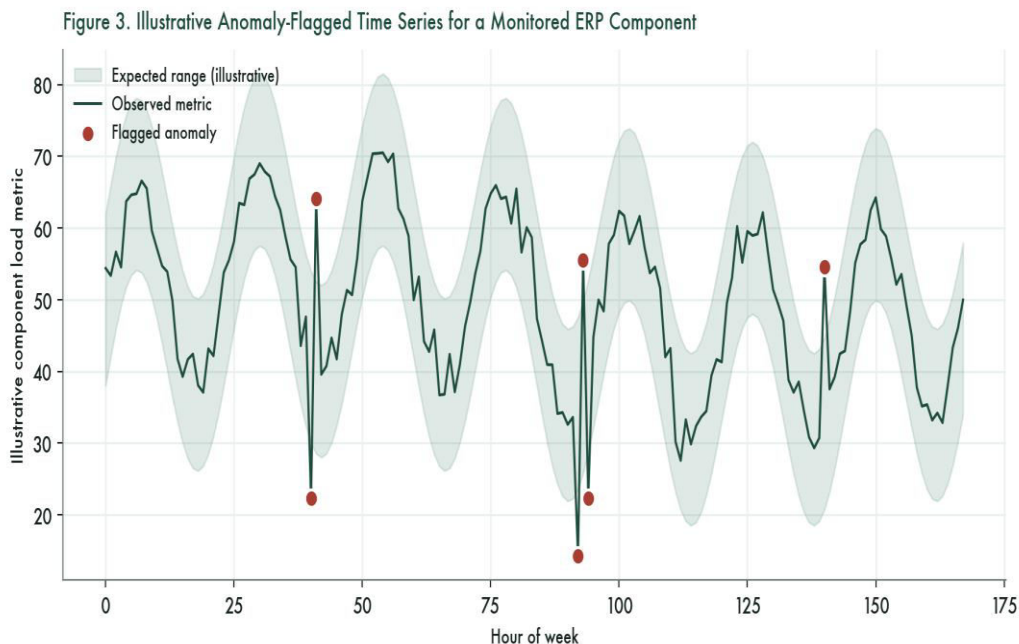


Figure.3. Illustrative anomaly-flagged time series for a monitored ERP component over one week, with an expected range band and six flagged anomalies

4.1 Model Comparison via ROC Curve

Figure 6 compares two illustrative candidate models - a gradient-boosted ensemble and a single decision tree - via receiver operating characteristic (ROC) curves, plotting true positive rate against false positive rate across all possible classification thresholds. The ensemble model's curve sits well above the single-tree model's across nearly the entire threshold range, reflected in its higher illustrative area-under-curve (AUC) value.

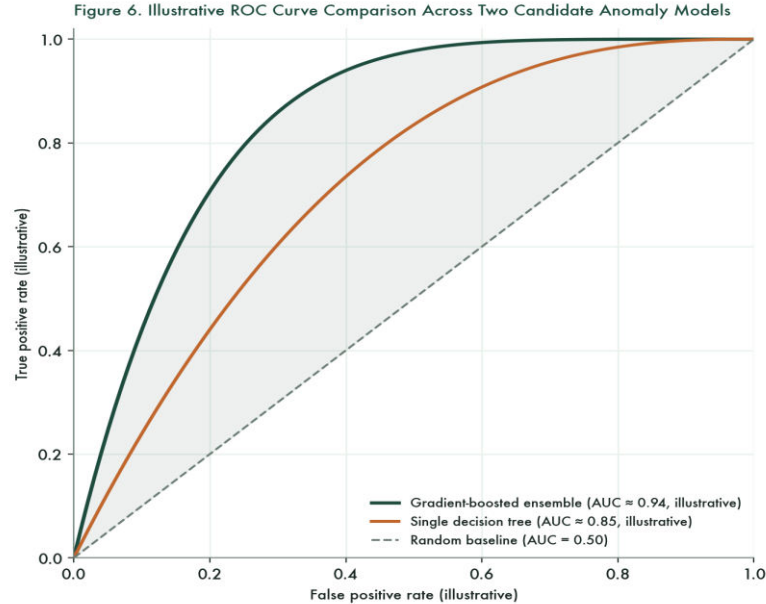


Figure.6. Illustrative ROC curve comparison between a gradient-boosted ensemble and a single decision tree candidate model, both compared against a random baseline.

4.2 Confusion Matrix on a Held-Out Test Set

Figure 7 presents an illustrative confusion matrix for the selected model evaluated against a held-out test set, decomposing outcomes into true negatives, false positives, false negatives, and true positives. This decomposition matters operationally because false positives and false negatives carry different costs: a false positive consumes on-call attention on a non-issue, while a false negative allows a genuine problem to proceed undetected.

Figure 7. Illustrative Confusion Matrix for the Anomaly Detection Model (Held-Out Test Set)



Figure.7. Illustrative confusion matrix for the anomaly detection model against a held-out test set, decomposed into the four standard outcome categories.

Table.2. Illustrative Evaluation Metrics Derived from the Confusion Matrix in Figure 7.

Metric	Formula (Illustrative Values)	What It Communicates
Precision	$126 / (126 + 38) \approx 0.77$	Of flagged anomalies, the share that were genuine
Recall (Sensitivity)	$126 / (126 + 24) \approx 0.84$	Of genuine anomalies, the share that were caught
Specificity	$912 / (912 + 38) \approx 0.96$	Of normal periods, the share correctly left unflagged
F1 Score	$2 \times (0.77 \times 0.84) / (0.77 + 0.84) \approx 0.80$	Harmonic mean balancing precision and recall

4.3 Choosing an Operating Threshold

Figure 8 presents an illustrative precision/recall trade-off curve across alert score thresholds, identifying a balanced operating point where the two curves cross. In practice, the appropriate operating threshold depends on organizational risk tolerance: an operations team with limited on-call capacity may deliberately choose a higher threshold favoring precision over recall, accepting some missed anomalies to avoid alert fatigue, while a team responsible for a compliance-sensitive component may choose a lower threshold favoring recall even at the cost of more false positives.

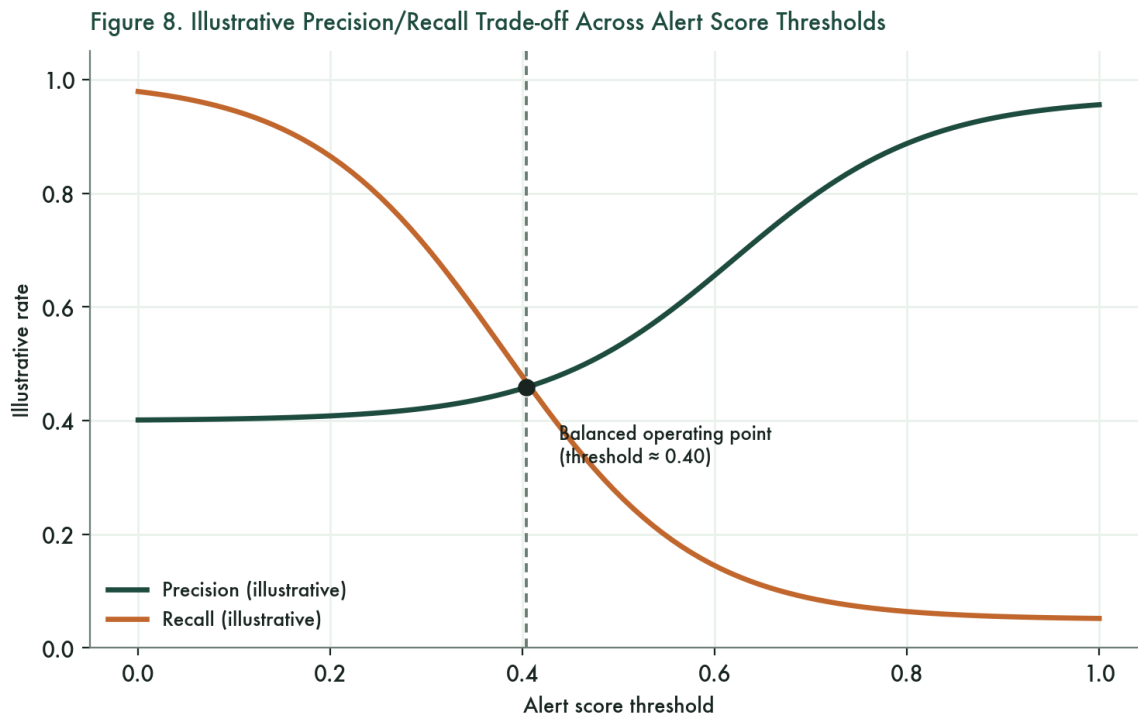


Figure.8. Illustrative precision/recall trade-off across alert score thresholds, identifying a balanced operating point.

V. CAPACITY FORECASTING WITH UNCERTAINTY QUANTIFICATION

Distinct from anomaly detection - which flags departures from expected behavior in the present or recent past - capacity forecasting projects a component's resource utilization forward in time, supporting proactive capacity planning rather than reactive incident response. Figure 4 presents an illustrative capacity forecast fan chart, showing thirty days of historical utilization followed by a thirty-day forecast with two confidence bands of increasing width.

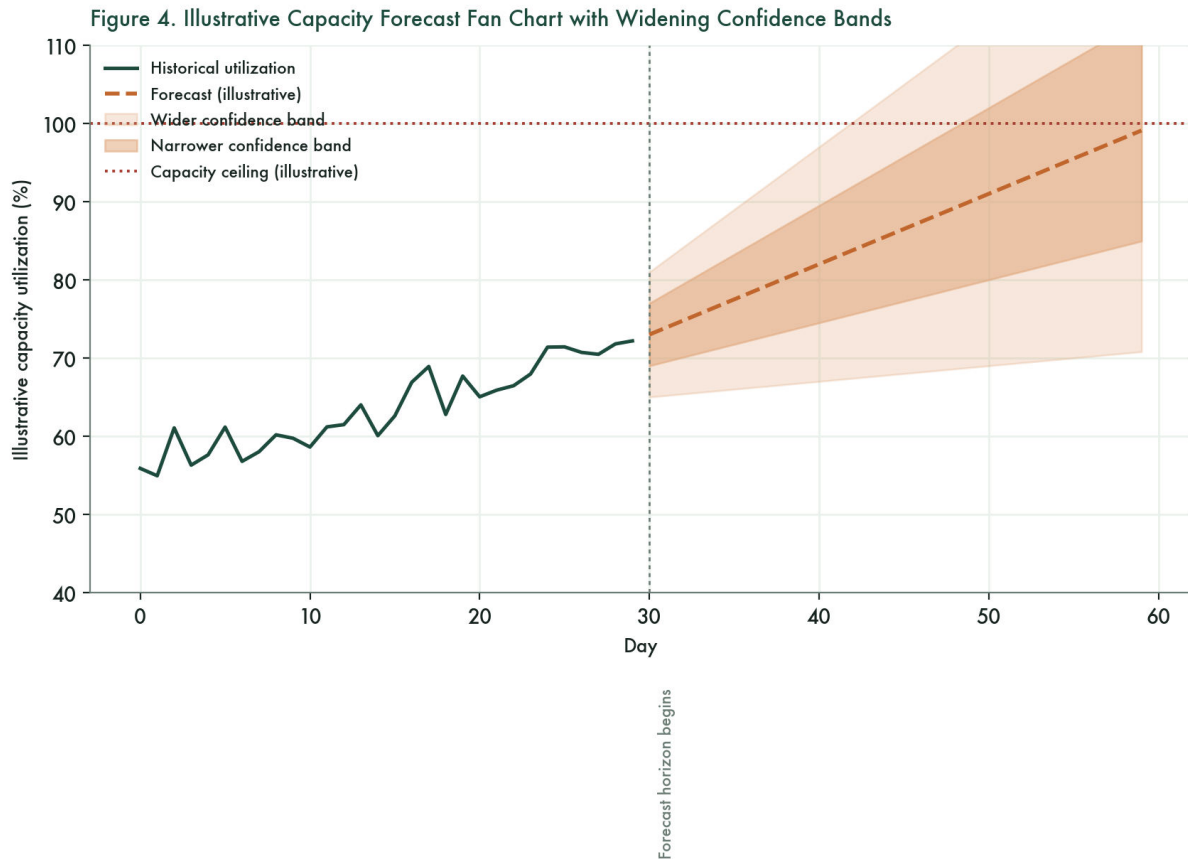


Figure.4. Illustrative capacity forecast fan chart showing historical utilization, a point forecast, and two widening confidence bands relative to an illustrative capacity ceiling.

5.1 Why Confidence Bands Widen Over the Forecast Horizon

The widening confidence bands in Figure 4 reflect a structural property of time-series forecasting: uncertainty compounds with forecast distance, since each additional day of forecast inherits the uncertainty of all preceding forecasted days plus its own. Presenting a forecast without confidence bands - a single point-estimate line - risks conveying false precision, particularly for capacity planning decisions with a multi-week or multi-month horizon where the band width by the end of the horizon may span a wide range of plausible outcomes.

5.2 Using the Capacity Ceiling as a Planning Trigger

The illustrative capacity ceiling line in Figure 4 represents the point at which a component would be expected to experience degraded performance or failure. A useful operational practice is to trigger a capacity planning review not when the point forecast crosses the ceiling, but when the lower bound of the narrower confidence band approaches it - providing lead time to plan a capacity expansion before the risk becomes acute, rather than waiting for the central estimate itself to signal a problem.

VI. FEATURE ENGINEERING FOR LANDSCAPE TELEMETRY

Figure 5 presents an illustrative feature importance ranking for the anomaly detection model described in Section 4, derived from the model's own internal importance scoring. CPU queue depth and batch job backlog count rank highest in this illustration, consistent with the intuition that resource contention signals tend to precede many categories of ERP performance incident.

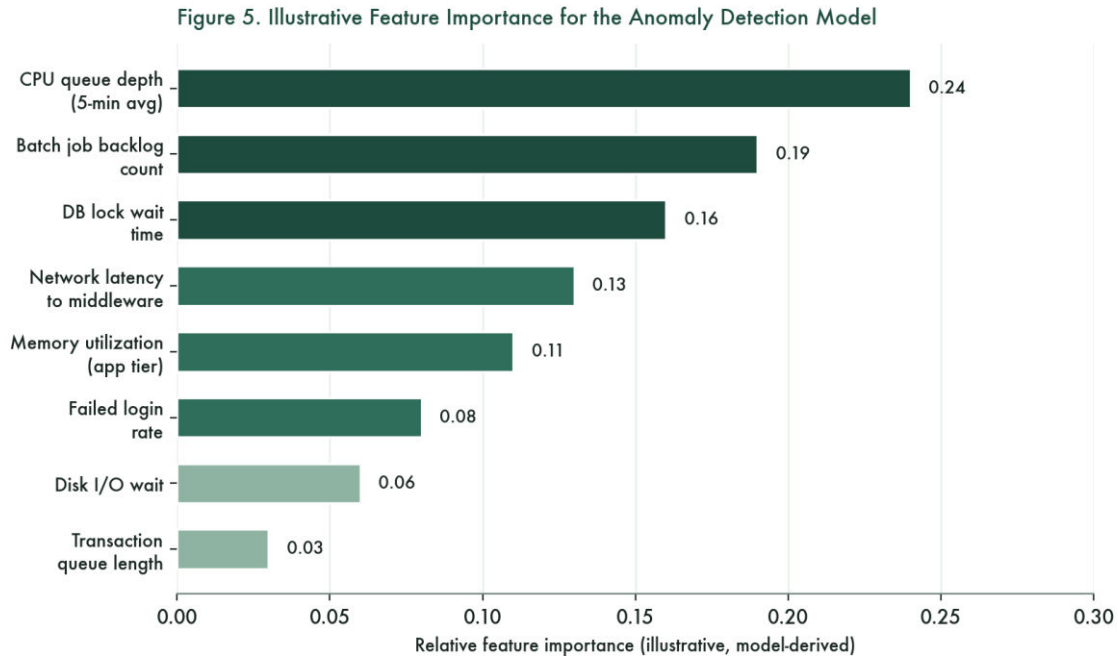


Figure.5. Illustrative feature importance ranking for the anomaly detection model, led by CPU queue depth and batch job backlog count

6.1 Windowing and Temporal Aggregation

Raw telemetry (a single CPU reading, a single queue depth reading) is rarely a useful direct model input; most effective features are aggregated over a rolling window - a 5-minute average, a 1-hour maximum, a day-over-day rate of change - chosen to match the timescale over which a given failure mode typically develops. Overly short windows produce noisy features that overfit to transient spikes; overly long windows smooth away the early signal that makes proactive detection valuable in the first place.

6.2 Cross-Component Features

As introduced in Section 2.2, features that capture relationships between components - a correlation coefficient between two components' load metrics over a recent window, or a lag indicating that Component A's degradation reliably precedes Component B's by a measurable interval - allow a model to distinguish an isolated local anomaly from an early symptom of a cascading landscape-wide issue, materially improving both detection lead time and root-cause attribution.

Table.3. Illustrative Feature Categories, Examples, and Typical Aggregation Windows

Feature Category	Example	Typical Aggregation Window
Resource utilization	CPU, memory, disk I/O wait	5-minute rolling average
Queue and backlog	Batch job backlog, transaction queue length	15-minute rolling average
Latency and response time	Network latency, DB lock wait time	5-minute p95
Rate-based	Failed login rate, error rate	1-hour rate of change
Cross-component	Correlation or lag between component pairs	1-hour to 24-hour rolling window

VII. FROM DETECTION TO ACTION: INCIDENT PREVENTION

A detection or forecast, on its own, prevents nothing; value is realized only when detection triggers timely and appropriate action. This section addresses the final pipeline stage in Figure 2 - automated and assisted response - and the design considerations that govern how much autonomy that stage should be granted.

7.1 A Three-Tier Response Model

1. **Tier 1 - Informational alert.** The system surfaces the detected anomaly or forecast risk to an operations dashboard or on-call channel, with no automated action taken. Appropriate for lower-confidence detections or components where an incorrect automated action carries meaningful risk.
2. **Tier 2 - Assisted response.** The system proposes a specific remediation action (restart a service, scale a resource pool, defer a non-critical batch job) and a human operator approves or rejects it before execution. Appropriate for moderate-confidence detections on components where the proposed action is well understood but carries some risk.
3. **Tier 3 - Automated response.** The system executes a predefined remediation action automatically, typically reserved for high-confidence detections on components where the remediation action is low-risk, well-tested, and reversible (for example, automatically scaling a stateless application server pool in response to a forecast capacity risk).

7.2 Matching Response Tier to Detection Confidence and Component Risk

The appropriate response tier for a given detection is a function of both the model's confidence in that specific detection and the component's risk profile - a high-confidence detection on a stateless, easily-recovered component may warrant Tier 3 automation, while even a high-confidence detection on a component touching financial postings or regulated data should typically remain at Tier 1 or Tier 2 given the consequence of an incorrect automated action.

VIII. COMPONENT-LEVEL BEHAVIOR PATTERNS

Figure 9 presents an illustrative distribution of capacity utilization across six component types using box plots, showing both central tendency and spread. The batch/job scheduler shows the highest median utilization and a notably wide distribution in this illustration, consistent with the bursty, schedule-driven load pattern characteristic of batch processing.

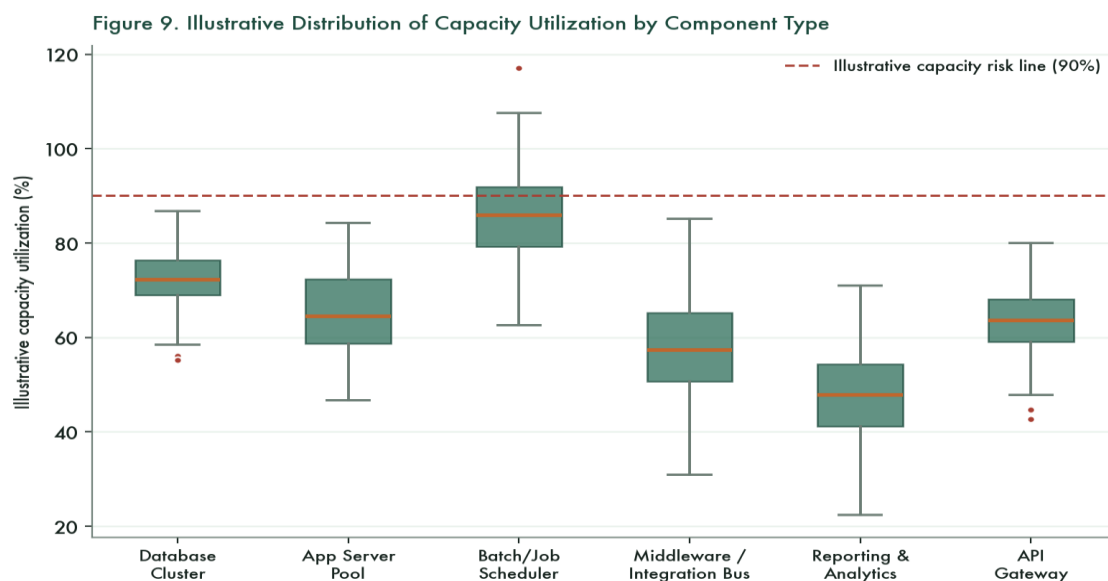


Figure.9. Illustrative distribution of capacity utilization by component type, showing the batch/job scheduler's high median and wide spread relative to other component types

8.1 Clustering Component Behavior

Figure 10 presents an illustrative two-feature projection of component behavior, showing distinct clusters corresponding to normal low-load and peak-load operation, alongside two visually separable anomalous regions - a latency-spike cluster and a set of isolated points representing resource-contention anomalies.

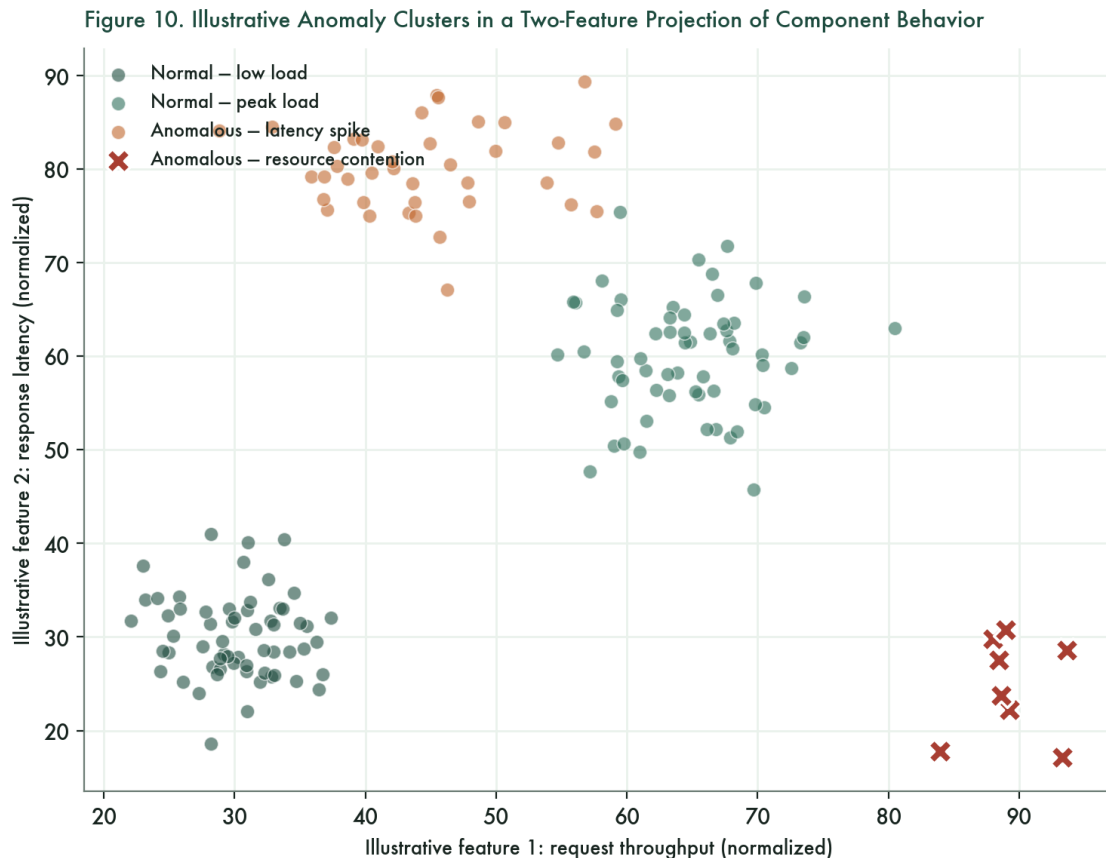


Figure.10. Illustrative anomaly clusters in a two-feature projection of component behavior, distinguishing normal operating clusters from two anomalous patterns.

Cluster-based views like Figure 10 are particularly useful for unsupervised anomaly detection, where labeled training examples of past anomalies are scarce: a new observation that falls outside all established normal-behavior clusters can be flagged as anomalous without requiring a prior labeled example of that specific failure mode, complementing the supervised approach detailed in Section 4.

IX. ILLUSTRATIVE OUTCOMES AND BEFORE/AFTER COMPARISON

Figure 11 presents an illustrative before/after comparison across five incident-related metrics, contrasting a pre-adoption baseline period with a post-adoption period following the rollout of the landscape intelligence capabilities described in this article.

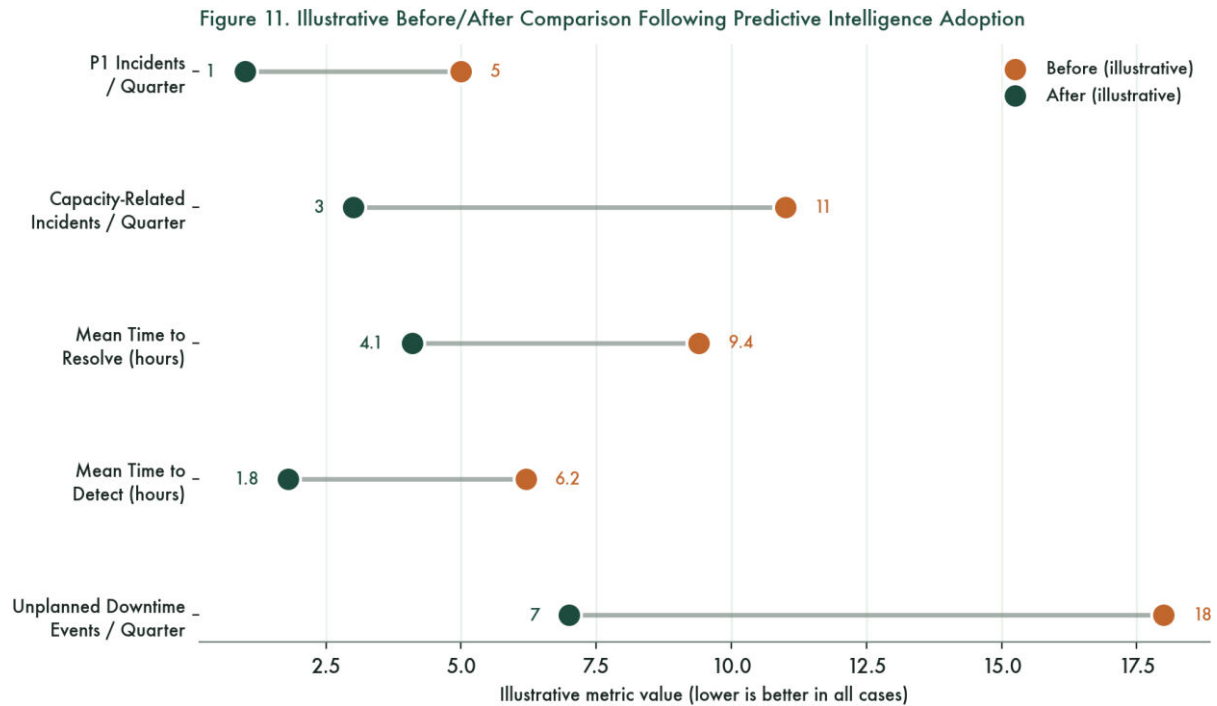


Figure.11. Illustrative before/after comparison across five incident-related metrics following predictive landscape intelligence adoption

The largest illustrative proportional improvement shown is in mean time to detect, consistent with the core value proposition of proactive anomaly detection: catching a developing issue materially earlier than a threshold-based alert would have fired. Capacity-related incidents also show a large illustrative improvement, consistent with the forecasting capability described in Section 5 enabling proactive capacity expansion ahead of a forecasted shortfall rather than reactive scrambling after a capacity-related outage.

Table.4. Illustrative Before/After Metrics Underlying Figure 11 (Estimated Figures, Not Measured Results).

Metric	Illustrative Before	Illustrative After	Relative Change
Unplanned downtime events / quarter	18	7	≈ 61% reduction
Mean time to detect (hours)	6.2	1.8	≈ 71% reduction
Mean time to resolve (hours)	9.4	4.1	≈ 56% reduction
Capacity-related incidents / quarter	11	3	≈ 73% reduction
P1 incidents / quarter	5	1	≈ 80% reduction

These before/after figures are explicitly illustrative and are constructed for methodological demonstration; they are not measured outcomes from a specific production deployment, and actual results will vary substantially based on landscape complexity, existing monitoring maturity, and implementation quality.

X. ADOPTION ROADMAP

Figure 12 presents an illustrative five-phase adoption roadmap, sequencing instrumentation and baselining ahead of model deployment, and separating anomaly detection deployment from capacity forecasting deployment given their distinct data and modeling requirements.

Figure 12. Illustrative Five-Phase Adoption Roadmap for Predictive Landscape Intelligence



Figure.12. Illustrative five-phase adoption roadmap: Instrument, Baseline, Detect, Forecast, and Automate.

10.1 Phase Descriptions

1. **Phase 1 - Instrument.** Deploy telemetry collection across all landscape components identified in the component health grid (Figure 1), establishing the data foundation every subsequent phase depends on.
2. **Phase 2 - Baseline.** Establish normal-behavior baselines for each component, capturing seasonal and cyclical patterns over a sufficient historical window - typically a minimum of several weeks to capture weekly cycles, and ideally a full quarter to capture month-end and quarter-end patterns relevant to ERP workloads specifically.
3. **Phase 3 - Detect.** Deploy anomaly detection models (Section 4), beginning with Tier 1 informational alerting (Section 7.1) before progressing to assisted or automated response.
4. **Phase 4 - Forecast.** Add capacity forecasting models (Section 5), typically sequenced after anomaly detection given the additional historical data volume forecasting models generally require to produce reliable confidence bands.
5. **Phase 5 - Automate.** Progress selected, well-understood response actions from Tier 1 or Tier 2 toward Tier 3 automation (Section 7.1), guided by the response-tier risk framework and informed by the feedback loop described in Section 3.1.

XI. RISKS, LIMITATIONS, AND OPEN QUESTIONS

11.1 Risks

- **Alert fatigue from miscalibrated thresholds.** An operating threshold set too aggressively (Section 4.3) generates enough false positives that operators begin ignoring alerts entirely, eroding the value of the entire detection capability regardless of underlying model quality.
- **Model drift as the landscape evolves.** As discussed in Section 3.1, a model trained on historical behavior can become progressively less accurate as the landscape changes, making the retraining feedback loop a governance necessity rather than an optional enhancement.
- **Over-automation risk.** Progressing a response action to Tier 3 automation (Section 7.1) before it has been sufficiently validated at Tier 2 risks an automated action causing more harm than the anomaly it was meant to address.

11.2 Limitations of This Article

This article synthesizes patterns from publicly available vendor documentation, AIOps and observability practice literature, and industry commentary current through October 2025. All model performance figures (Sections 4 and 5), before/after outcomes (Section 9), and KPI figures presented on the cover are explicitly illustrative and constructed to demonstrate method rather than to report measured results from a specific organization's deployment.

11.3 Open Questions

- How should cross-component feature engineering (Section 6.2) scale as landscape component count grows well beyond the twelve illustrated in this article, given that cross-component feature pairs grow combinatorially with component count?
- What governance process should determine when a response action is sufficiently validated to progress from Tier 2 assisted response to Tier 3 full automation (Section 7.1), and how should that determination be documented for audit purposes?
- As ERP vendors increasingly ship first-party AIOps and landscape intelligence capabilities, how should independently built models of the kind described in this article be positioned relative to vendor-native offerings - as a complement, a more customizable alternative, or an interim capability pending vendor feature maturity?

XII. CONCLUSION

Predictive landscape intelligence represents a shift from reactive, threshold-based ERP monitoring toward a proactive discipline grounded in learned models of normal component behavior and forward-looking capacity forecasts. The reference pipeline, evaluation methodology, and response-tier framework presented in this article are offered as a practical starting point for organizations building this capability. As with any machine learning system operating in a consequential domain, success depends less on initial model accuracy than on the surrounding operational disciplines - a functioning feedback loop, a deliberate response-tier progression, and honest evaluation against held-out data - that keep the system's judgment trustworthy as the landscape it monitors continues to evolve.

REFERENCES

- [1] Amazon Web Services. "Amazon Lookout for Metrics: Anomaly Detection Documentation." AWS Documentation, 2024.
- [2] Amazon Web Services. "Forecasting at Scale with Amazon Forecast." AWS Machine Learning Blog, 2024.
- [3] Bishop, C. M. "Pattern Recognition and Machine Learning." Springer, 2006.
- [4] Box, G. E. P., Jenkins, G. M., Reinsel, G. C., and Ljung, G. M. "Time Series Analysis: Forecasting and Control," 5th Edition. Wiley, 2015.
- [5] Chandola, V., Banerjee, A., and Kumar, V. "Anomaly Detection: A Survey." ACM Computing Surveys, 2009.
- [6] Datadog, Inc. "The State of AIOps and Observability 2025." Datadog Research, 2025.
- [7] Dynatrace LLC. "Davis AI: Causal AI for IT Operations." Dynatrace Documentation, 2024.
- [8] Gartner, Inc. "Market Guide for AIOps Platforms." Gartner Research, 2024.
- [9] Gartner, Inc. "Predicts 2025: AI in IT Operations and Observability." Gartner Research, 2024.
- [10] Google Cloud. "Vertex AI Forecasting: Technical Overview." Google Cloud Documentation, 2024.
- [11] Hochreiter, S., and Schmidhuber, J. "Long Short-Term Memory." Neural Computation, 1997.
- [12] Hyndman, R. J., and Athanasopoulos, G. "Forecasting: Principles and Practice," 3rd Edition. OTexts, 2021.
- [13] IBM Corporation. "IBM Instana Observability: AI-Powered Anomaly Detection." IBM Documentation, 2024.
- [14] Microsoft Corporation. "Azure Monitor: Dynamic Thresholds and Smart Detection." Microsoft Learn, 2024.
- [15] New Relic, Inc. "Applied Intelligence: Anomaly Detection Documentation." New Relic Documentation, 2024.
- [16] Provost, F., and Fawcett, T. "Data Science for Business." O'Reilly Media, 2013.
- [17] SAP SE. "SAP Focused Run: AI-Based Anomaly Detection for SAP Landscapes." SAP Documentation, 2024.
- [18] SAP SE. "SAP HANA Predictive Analysis Library (PAL) Reference." SAP Help Portal, 2024.
- [19] Siegel, E. "Predictive Analytics: The Power to Predict Who Will Click, Buy, Lie, or Die," Revised Edition. Wiley, 2016.
- [20] Splunk Inc. "Splunk IT Service Intelligence: Predictive Analytics Documentation." Splunk Documentation, 2024.
- [21] Taylor, S. J., and Letham, B. "Forecasting at Scale (Prophet)." The American Statistician, 2018.

- [22] The Linux Foundation / CNCF. "OpenTelemetry Specification." CNCF Documentation, 2024.
- [23] Witten, I. H., Frank, E., Hall, M. A., and Pal, C. J. "Data Mining: Practical Machine Learning Tools and Techniques," 4th Edition. Morgan Kaufmann, 2016.
- [24] World Economic Forum. "Global Risks Report 2025: Technology and Infrastructure Resilience." WEF Publications, 2025.
- [25] Zhou, Z.-H. "Machine Learning." Springer, 2021.

Appendix A: Glossary of Terms

AIOps - The application of artificial intelligence and machine learning techniques to IT operations tasks, including anomaly detection, root-cause analysis, and automated remediation.

Anomaly Detection - The identification of data points, events, or patterns that deviate significantly from expected or normal behavior.

AUC (Area Under the Curve) - A single summary metric derived from a ROC curve, representing the probability that a model ranks a randomly chosen positive example higher than a randomly chosen negative example; ranges from 0.5 (random) to 1.0 (perfect).

Confusion Matrix - A table summarizing a classification model's predictions against actual outcomes, decomposed into true positives, true negatives, false positives, and false negatives.

ERP (Enterprise Resource Planning) - Integrated software managing core business processes such as finance, procurement, supply chain, and human resources within a single system of record.

Feature Engineering - The process of transforming raw data into input variables (features) that better represent the underlying problem to a predictive model.

Precision - Of all instances a model predicted as positive (e.g., anomalous), the proportion that were actually positive.

Recall (Sensitivity) - Of all instances that were actually positive, the proportion the model correctly identified as positive.

ROC Curve (Receiver Operating Characteristic) - A plot of true positive rate against false positive rate across all classification thresholds, used to evaluate and compare binary classifiers.

Time-Series Forecasting - The use of historical, time-ordered data to predict future values of a metric, typically with an associated measure of forecast uncertainty.



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details