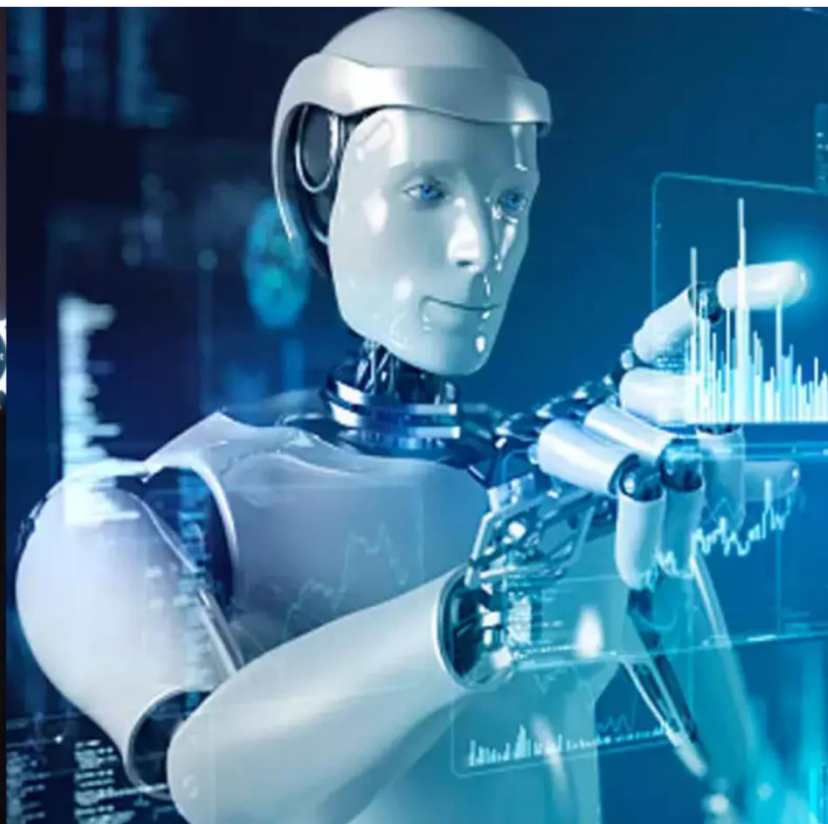




International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

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



Impact Factor: 8.699

Volume 15, Issue 2, February 2026

Autonomous ERP Operations: Designing Self-Healing Infrastructure Pipelines using Generative AI for Real-Time Diagnosis, Root Cause Analysis, and Remediation in Mission-Critical Landscapes

Srinivas Kakarla

SAP Architect, USA

ABSTRACT: Mission-critical enterprise resource planning (ERP) landscapes have traditionally relied on human operators to close the loop between detecting an infrastructure issue and resolving it - a model that struggles to keep pace with the scale, interdependency, and around-the-clock availability expectations of modern ERP operations. This article presents an architecture for self-healing infrastructure pipelines that use generative AI (GenAI) to perform real-time diagnosis and root cause analysis, and to plan and, within defined guardrails, execute remediation autonomously. We introduce a five-stage self-healing loop (Sense, Diagnose, Decide, Act, Verify), a layered GenAI reasoning architecture grounded in retrieval-augmented generation over runbooks and topology graphs, a six-level autonomy ladder for governing how much independence the system is granted, and a blast-radius containment model for bounding the scope of autonomous action. All illustrative time and cost figures presented - including mean-time-to-resolve estimates - are explicitly flagged as estimates for methodological demonstration rather than measured results from a specific production deployment.

KEYWORDS: Enterprise Resource Planning, Generative AI, Self-Healing Infrastructure, Root Cause Analysis, Autonomous Remediation, AIOps, Retrieval-Augmented Generation, Incident Response, Mission-Critical Systems, Blast Radius.

I. INTRODUCTION: FROM ALERTING TO SELF-HEALING

Mission-critical ERP landscapes - those underpinning financial close, order-to-cash, and regulated supply chain processes - have historically treated infrastructure resilience as a human-mediated discipline: an alert fires, an on-call engineer investigates, a root cause is identified, and a remediation is applied, each step requiring human judgment and hand-off. This model has served reliably for decades, but it faces mounting strain as ERP landscapes grow more distributed, more interdependent, and more continuously available by business expectation.

Generative AI (GenAI) - specifically, large language models capable of reasoning over unstructured context such as runbooks, incident histories, and system topology - introduces a new capability at the diagnosis and decision stage of this workflow: the ability to synthesize telemetry, historical precedent, and operational documentation into a root-cause hypothesis and a proposed remediation plan, in something close to real time. This article presents an architecture for embedding that capability into a self-healing infrastructure pipeline, structured around a five-stage operational loop and a graduated model of autonomy.

Self-healing is not a binary switch from manual to automated operations - it is a spectrum of graduated autonomy, and the organizations that succeed with it are the ones that make that spectrum explicit rather than implicit.

The remainder of this article proceeds as follows. Section 2 presents the self-healing loop. Section 3 details a layered GenAI reasoning architecture. Section 4 covers the remediation decision tree that governs autonomy tier selection. Section 5 introduces a six-level autonomy ladder. Section 6 addresses blast-radius containment. Section 7 discusses dependency graphs for root-cause reasoning. Section 8 presents illustrative outcomes. Section 9 examines seasonal incident patterns. Section 10 offers implementation considerations. Section 11 addresses risks and open questions, and Section 12 concludes.

II. THE SELF-HEALING LOOP

Figure 1 presents the self-healing loop as a five-stage cycle - Sense, Diagnose, Decide, Act, Verify - with a human oversight gate positioned at the center, reachable from any stage when confidence is insufficient or the proposed action exceeds the autonomy boundaries described in Section 5.

Figure 1. The Self-Healing Loop: Sense - Diagnose - Decide - Act - Verify



Figure.1. The self-healing loop: Sense, Diagnose, Decide, Act, Verify, with a central human oversight gate reachable from every stage.

2.1 Stage Descriptions

- 1. Sense.** Continuous ingestion of telemetry, logs, traces, and change events across the ERP landscape, providing the raw signal from which anomalies and incidents are detected.
- 2. Diagnose.** GenAI-driven root-cause reasoning, synthesizing current telemetry against historical incident patterns, system topology, and operational runbooks (detailed in Section 3).
- 3. Decide.** Formulation of a remediation plan paired with a confidence score, which - together with the incident's blast radius - determines the appropriate autonomy tier (Section 4).

4. **Act.** Execution of the remediation plan, either fully automated, human-approved, or purely informational, depending on the autonomy tier selected in the Decide stage.
5. **Verify.** Confirmation that the applied remediation resolved the underlying issue, with the outcome logged to inform future diagnosis and, where applicable, model retraining.

2.2 Why the Human Oversight Gate Sits at the Center, Not the End

A common design mistake is placing human oversight only at the end of the loop, reviewing outcomes after remediation has already been applied. The architecture in Figure 1 instead positions the oversight gate at the center, reachable from the Decide stage before action is taken whenever the confidence and blast-radius criteria (Section 4) call for it - ensuring human judgment is available before a consequential action, not only after.

III. A LAYERED GENAI REASONING ARCHITECTURE

Figure 2 presents a four-layer architecture underpinning the Diagnose and Act stages of the self-healing loop: a telemetry layer providing raw signal, a context and retrieval layer that grounds the GenAI reasoning process in relevant runbooks, topology data, and historical incidents (a retrieval-augmented generation, or RAG, pattern), a GenAI reasoning layer that generates root-cause hypotheses with confidence scores, and a remediation orchestration layer that executes approved actions with rollback control.

Figure 2. Layered GenAI Diagnosis and Remediation Architecture for Autonomous ERP Operations

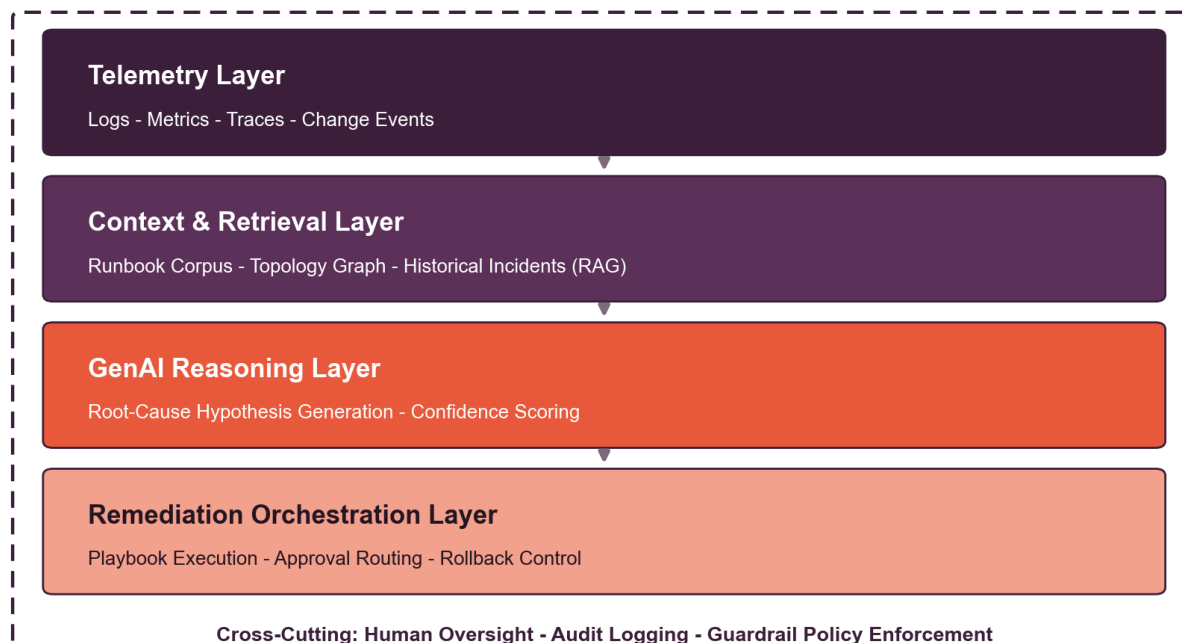


Figure.2. Layered GenAI diagnosis and remediation architecture, with cross-cutting human oversight, audit logging, and guardrail policy enforcement spanning all four layers

3.1 Why Retrieval-Augmented Generation Matters Here

A GenAI model reasoning purely from its trained parameters, without grounding in an organization's specific runbooks, topology, and incident history, risks generating a plausible-sounding but incorrect root-cause hypothesis - a failure mode with real consequence when the hypothesis feeds directly into an automated remediation action. The context and retrieval layer in Figure 2 addresses this by retrieving the most relevant runbook entries, topology relationships, and precedent incidents at inference time, grounding the reasoning layer's output in the specific landscape it is diagnosing rather than generic prior knowledge.

3.2 Rollback Control as a First-Class Design Element

The remediation orchestration layer's inclusion of rollback control is deliberate: any automated action taken by the system should have a corresponding, tested rollback path, so that an incorrect remediation can itself be quickly reversed rather than compounding the original incident. This is analogous to the transaction-rollback guarantees ERP systems already provide at the data layer, extended here to the operational-action layer.

IV. FROM DIAGNOSIS TO DECISION: THE REMEDIATION DECISION TREE

Figure 4 presents an illustrative decision tree governing which autonomy tier a given anomaly is routed to, based on the GenAI reasoning layer's confidence score and the incident's estimated blast radius (Section 6).

Figure 4. Illustrative Remediation Decision Tree: Confidence and Blast Radius Drive Autonomy Tier

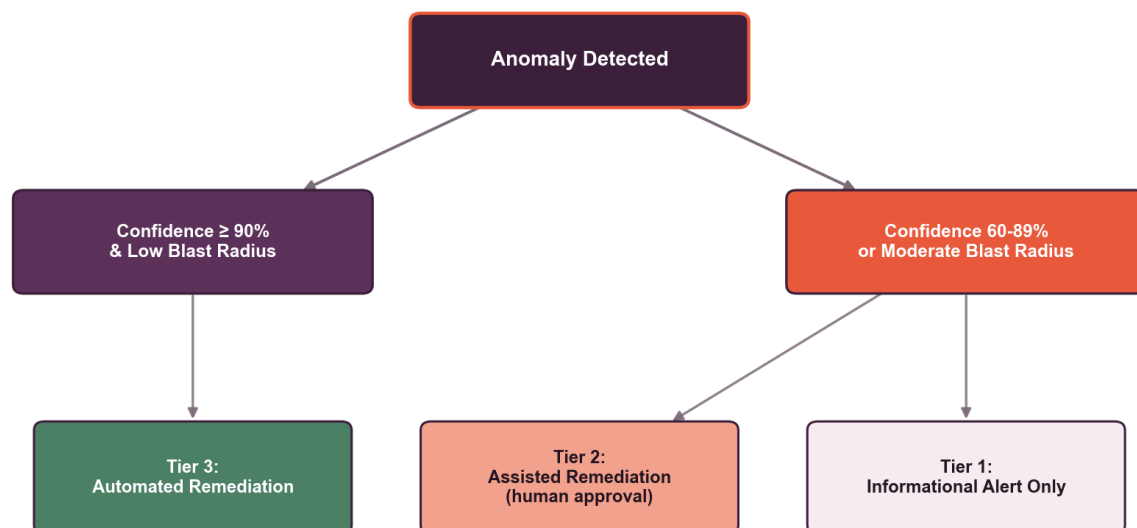


Figure.4. Illustrative remediation decision tree: high confidence and low blast radius route to automated remediation; lower confidence or wider blast radius route to assisted remediation or informational alerting only.

Autonomy Tier	Confidence Threshold	Blast Radius	Human Involvement
Tier 3: Automated	$\geq 90\%$	Low (service-level)	None at execution; sampled post-hoc audit
Tier 2: Assisted	60–89%	Moderate (module-level)	Approval required before execution
Tier 1: Informational	$< 60\%$ or high blast radius	High (landscape-wide)	Full manual diagnosis and remediation

Table 1. Autonomy Tier Assignment Criteria Based on Confidence Score and Blast Radius.

This decision structure ensures that the system's most consequential actions - those with the widest potential blast radius - always retain a human decision point, while routine, well-understood, narrowly-scoped issues can be resolved without incurring the latency and cognitive cost of human review.

V. THE AUTONOMY-LEVEL LADDER

Beyond the per-incident tiering in Section 4, organizations adopting self-healing infrastructure benefit from an explicit maturity model describing how much autonomy the overall system is granted at a given point in its adoption journey. Figure 6 presents a six-level autonomy ladder, from fully manual operations (L0) to full autonomy (L5).

Figure 6. The Autonomy-Level Ladder for Self-Healing ERP Infrastructure (L0-L5)

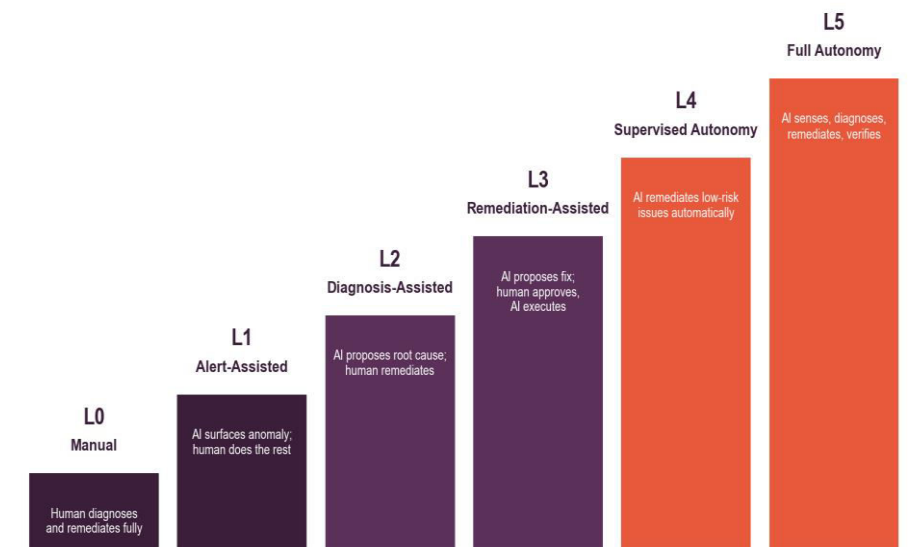


Figure.6. The autonomy-level ladder for self-healing ERP infrastructure, from L0 (Manual) through L5 (Full Autonomy)

Table.2. The Six Levels of the Autonomy Ladder and Their Operational Definitions

Level	Name	Description
L0	Manual	Human diagnoses and remediates the issue fully, without AI assistance.
L1	Alert-Assisted	AI surfaces the anomaly; a human performs all diagnosis and remediation.
L2	Diagnosis-Assisted	AI proposes a root cause; a human performs remediation.
L3	Remediation-Assisted	AI proposes a fix; a human approves, and the AI executes it.
L4	Supervised Autonomy	AI remediates low-risk issues automatically, with sampled human audit.
L5	Full Autonomy	AI senses, diagnoses, remediates, and verifies without human intervention for in-scope issue classes.

5.1 Progressing Up the Ladder Is a Governance Decision, Not Only a Technical One

Advancing from one autonomy level to the next should be a deliberate governance decision, supported by a track record of accuracy at the current level, rather than a default outcome of model capability alone. An organization may reach L5-capable model accuracy for a given issue class well before organizational risk tolerance and audit requirements are ready to grant L5 autonomy for that class in production.

VI. BLAST-RADIUS CONTAINMENT

Figure 7 presents a concentric blast-radius containment model, distinguishing service-level, module-level, and landscape-wide scope around a given root cause. This model is the second input (alongside confidence score) to the autonomy-tier decision in Section 4.

Figure 7. Illustrative Blast-Radius Containment Model for Autonomous Remediation Scope

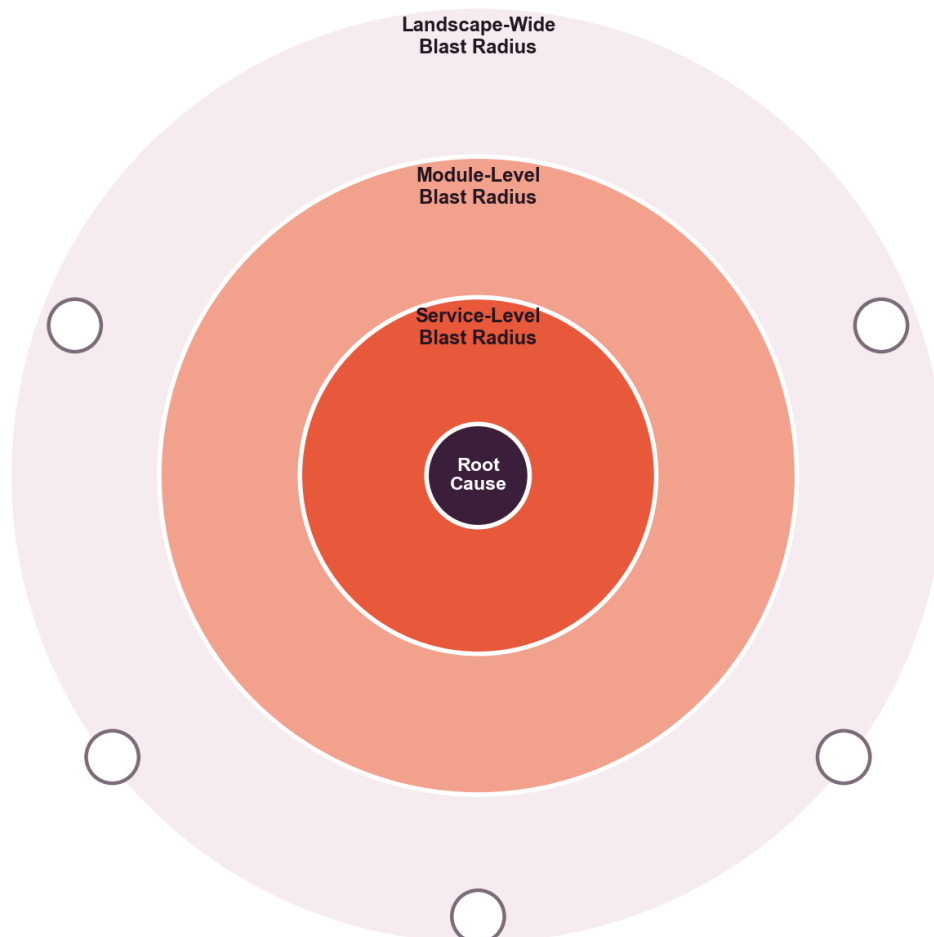


Figure.7. Illustrative blast-radius containment model, distinguishing service-level, module-level, and landscape-wide scope around a root cause.

Estimating blast radius accurately depends heavily on an accurate, current component dependency graph (Section 7); an out-of-date topology model risks underestimating blast radius and granting excessive autonomy to a remediation that, in practice, has landscape-wide implications the model was not aware of.

VII. COMPONENT DEPENDENCY GRAPHS FOR ROOT-CAUSE REASONING

Figure 9 presents an illustrative dependency graph spanning seven landscape components, rooted at a middleware bus that mediates communication between the ERP core modules and the underlying database, application server, and identity services.

Figure 9. Illustrative Component Dependency Graph Used for Blast-Radius and Root-Cause Reasoning



Figure.9. Illustrative component dependency graph used for blast-radius and root-cause reasoning, rooted at the middleware bus as the highest-connectivity component.

7.1 Why the Highest-Connectivity Component Deserves the Most Conservative Autonomy Policy

As Figure 9 illustrates, the middleware bus has the highest connectivity of any component in this landscape - a failure or misconfigured remediation there has the broadest potential blast radius. Organizations should apply their most conservative autonomy policy (favoring Tier 1 or Tier 2 per Section 4) to the highest-connectivity components in their own dependency graph, reserving Tier 3 automated remediation for leaf or near-leaf components where a remediation action's consequences are more naturally contained.

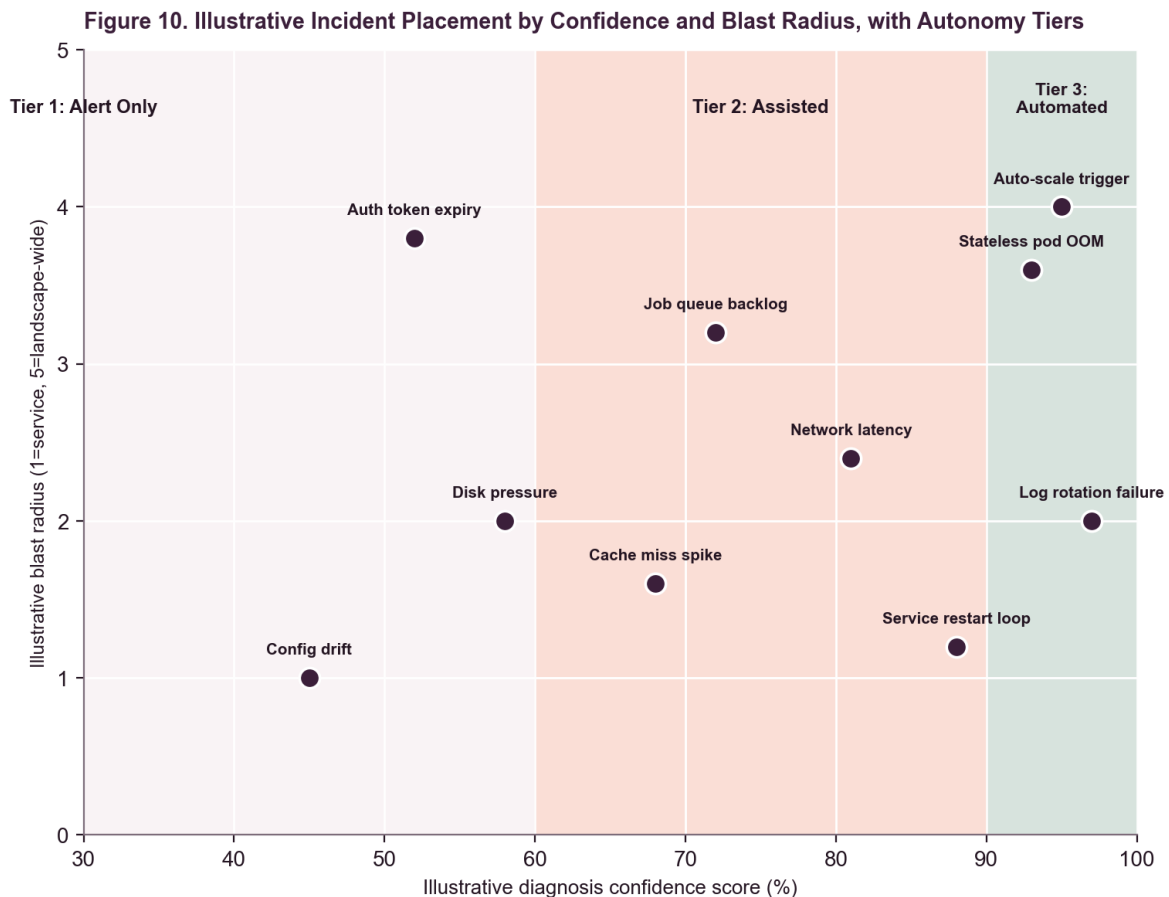


Figure.10. Illustrative placement of ten example incidents by diagnosis confidence and blast radius, shown against the three autonomy-tier regions from Section 4

Figure 10 places ten illustrative example incidents on the same confidence and blast-radius axes used for tier assignment, showing how a mix of incident types - from configuration drift to stateless pod memory exhaustion - distributes across the three autonomy tiers in practice.

VIII. ILLUSTRATIVE OUTCOMES: MTTR AND INCIDENT SEVERITY TRENDS

Figure 5 presents an illustrative mean-time-to-resolve (MTTR) waterfall, decomposing an estimated reduction from a 180-minute baseline to a 35-minute resulting MTTR across the contributing self-healing pipeline stages: faster detection, automated diagnosis, automated remediation, and reduced escalation hops.

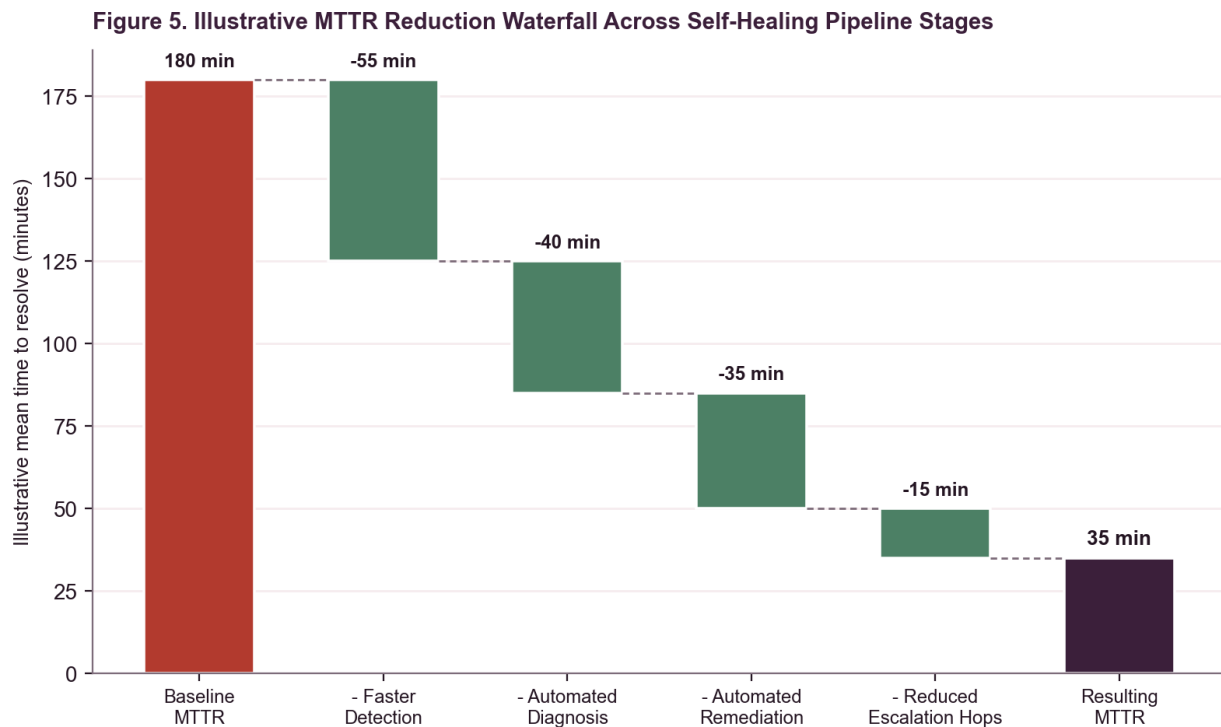


Figure.5. Illustrative MTTR reduction waterfall, decomposing time savings across four contributing self-healing pipeline stages

\$ ESTIMATE - The MTTR figures in Figure 5 (180 minutes baseline, 35 minutes resulting, and the intermediate stage reductions) are illustrative estimates for methodological demonstration only. They are not measured results from a specific production deployment and will vary by organization.

Figure 3 presents an illustrative ten-month trend in incident severity following a self-healing pipeline rollout, showing Sev-1 (critical), Sev-2 (moderate), and Sev-3 (low) incident counts declining across the period, consistent with a maturing self-healing capability progressively absorbing more issue classes at higher autonomy tiers.

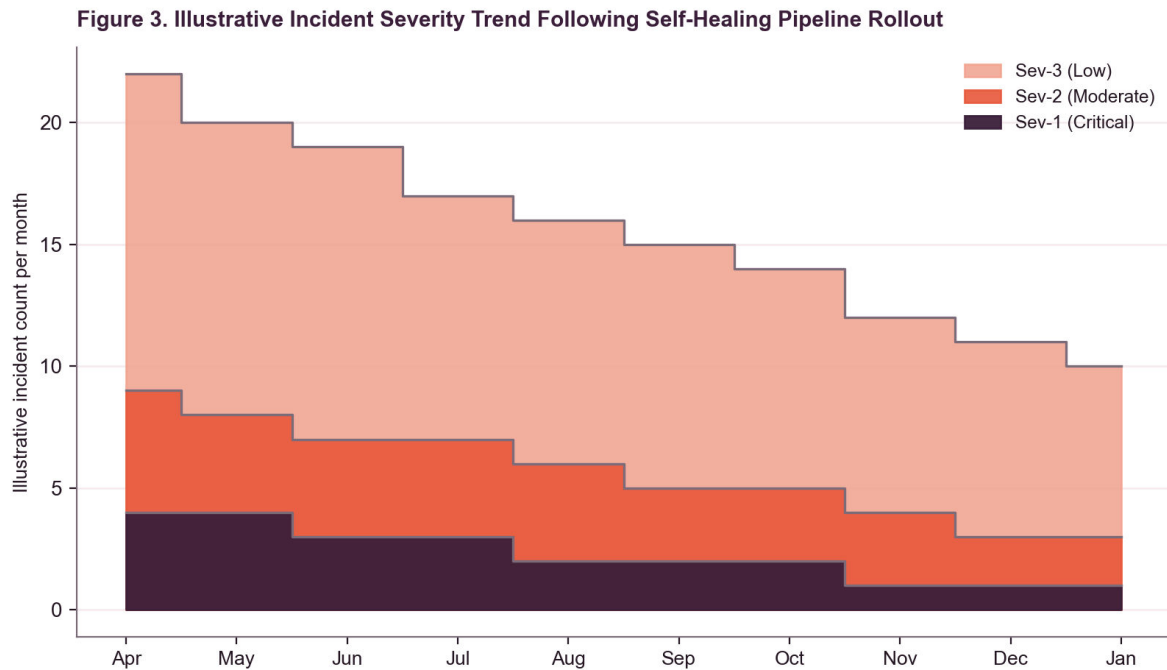


Figure.3. Illustrative incident severity trend across ten months following self-healing pipeline rollout, showing declining counts across all three severity bands.

Table.3. Summary of Illustrative Outcome Metrics Underlying Figures 3 and 5 (Estimated Figures)

Metric	Illustrative Estimated Value	Basis
Baseline MTTR	180 minutes	Pre-rollout estimate
Resulting MTTR	35 minutes	Post-rollout estimate, $\approx$ 81% illustrative reduction
Sev-1 incidents, Month 1 vs. Month 10	4 vs. 1	$\approx$ 75% illustrative reduction
Sev-3 incidents, Month 1 vs. Month 10	22 vs. 10	$\approx$ 55% illustrative reduction

\$ ESTIMATE - All dollar figures referenced in this section are illustrative estimates for discussion purposes only. They are not drawn from a specific organization's actual costs and should not be used as a benchmark or budgeting input without independent validation.

IX. SEASONAL AND CATEGORY PATTERNS IN MISSION-CRITICAL INCIDENTS

Figure 11 presents an illustrative twelve-month heatmap of incident counts across five categories, surfacing patterns that a simple monthly aggregate would obscure - notably, elevated batch/job failure counts at illustrative quarter-end months and elevated capacity saturation incidents toward year-end.

Figure 11. Illustrative Seasonal Incident Heatmap by Category and Month

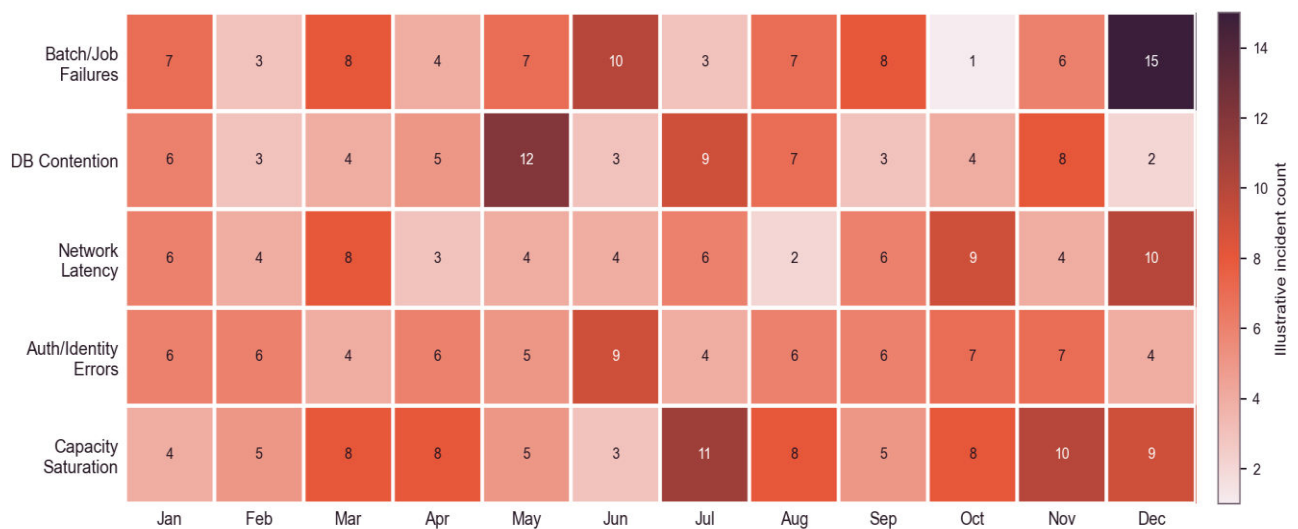


Figure.11. Illustrative seasonal incident heatmap by category and month, showing quarter-end batch spikes and year-end capacity saturation patterns

9.1 Using Seasonal Patterns to Pre-Position Autonomy

A mature self-healing program can use patterns like those in Figure 11 proactively: if batch/job failures reliably spike at quarter-end, an organization might deliberately grant Tier 3 automated remediation to well-understood batch-failure issue classes ahead of the quarter-end window, rather than waiting for volume to build before considering autonomy expansion - provided the confidence and blast-radius criteria from Section 4 are otherwise satisfied.

X. IMPLEMENTATION CONSIDERATIONS

1. **Start at Tier 1 for every new issue class.** Even when a GenAI model shows strong illustrative confidence for a novel issue class, initial production exposure should begin at Tier 1 (informational only) to build a validated accuracy track record before progressing.
2. **Keep the dependency graph current.** As emphasized in Section 6, blast-radius estimation is only as good as the underlying topology data; stale dependency graphs are a leading cause of underestimated blast radius in practice.
3. **Invest in the retrieval corpus, not just the model.** As discussed in Section 3.1, the quality of runbook and historical-incident content available to the retrieval layer often matters more to diagnostic accuracy than incremental improvements to the underlying GenAI model.
4. **Log every autonomous action for audit.** The cross-cutting audit logging layer in Figure 2 is not optional overhead - it is the evidence base for both the sampled post-hoc audits at Tier 3 (Table 1) and for justifying autonomy-ladder progression decisions (Section 5.1).

XI. RISKS, LIMITATIONS, AND OPEN QUESTIONS

11.1 Risks

- **Hallucinated root-cause hypotheses.** A GenAI reasoning layer insufficiently grounded by retrieval (Section 3.1) can generate a confident-sounding but incorrect diagnosis, which is especially dangerous if that diagnosis feeds directly into a Tier 3 automated remediation.
- **Blast-radius underestimation.** As discussed in Section 6, an inaccurate or stale dependency graph can lead the system to grant more autonomy than an incident's true scope warrants.
- **Autonomy-ladder overreach.** Organizational pressure to demonstrate automation value can create incentive to progress up the autonomy ladder (Section 5) faster than the underlying accuracy track record supports.

11.2 Limitations of This Article

This article synthesizes patterns from publicly available vendor documentation, AIOps and site reliability engineering practice literature, and industry commentary current through January 2026. All MTTR, severity-trend, and cost-related figures presented are explicitly illustrative estimates constructed to demonstrate method, not measured outcomes from a specific production deployment.

11.3 Open Questions

- How should the confidence score itself be validated and calibrated over time, given that a miscalibrated confidence score undermines the entire tiering framework in Section 4?
- What audit and regulatory standards will emerge specifically for autonomous remediation actions in mission-critical financial and supply-chain systems, and how will they interact with existing internal-control frameworks?
- As ERP vendors ship increasingly capable first-party self-healing features, how should independently built architectures of the kind described in this article be positioned relative to vendor-native offerings?

XII. CONCLUSION

Self-healing infrastructure for mission-critical ERP landscapes is best understood as a graduated capability, not a destination. The self-healing loop, layered GenAI architecture, autonomy ladder, and blast-radius containment model presented in this article are offered as a shared vocabulary for building that capability deliberately - expanding autonomy in step with demonstrated accuracy and organizational risk tolerance, rather than treating automation as a binary switch. Organizations that invest as much in their retrieval corpus, dependency graphs, and audit logging as they do in the underlying GenAI model will be best positioned to expand autonomy safely as their self-healing program matures.

REFERENCES

- [1] Amazon Web Services. "Amazon Bedrock Agents: Automating Multi-Step Tasks." AWS Documentation, 2024.
- [2] Anthropic. "Building Effective Agents." Anthropic Engineering Blog, 2024.
- [3] Beyer, B., Jones, C., Petoff, J., and Murphy, N. R. "Site Reliability Engineering: How Google Runs Production Systems." O'Reilly Media, 2016.
- [4] Beyer, B., Murphy, N. R., Rensin, D. K., Kawahara, K., and Thorne, S. "The Site Reliability Workbook." O'Reilly Media, 2018.
- [5] Datadog, Inc. "The State of AIOps and Observability 2025." Datadog Research, 2025.
- [6] Dynatrace LLC. "Davis AI: Causal AI for IT Operations." Dynatrace Documentation, 2024.
- [7] Gartner, Inc. "Predicts 2025: The Rise of Agentic AI in the Enterprise." Gartner Research, 2024.
- [8] Gartner, Inc. "Market Guide for AIOps Platforms." Gartner Research, 2024.
- [9] GitHub, Inc. "Copilot Workspace: From Issue to Pull Request." GitHub Blog, 2024.
- [10] IBM Corporation. "IBM Instana Observability: AI-Powered Anomaly Detection." IBM Documentation, 2024.
- [11] Kim, G., Humble, J., Debois, P., and Willis, J. "The DevOps Handbook," 2nd Edition. IT Revolution Press, 2021.
- [12] Lewis, P., et al. "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks." Advances in Neural Information Processing Systems (NeurIPS), 2020.
- [13] Limoncelli, T. A., Chalup, S. R., and Hogan, C. J. "The Practice of Cloud System Administration," Volume 2. Addison-Wesley, 2015.

- [14] Microsoft Corporation. "Autonomous Agents in Copilot Studio: Architecture Overview." Microsoft Learn, 2024.
- [15] Microsoft Corporation. "Azure Monitor: Dynamic Thresholds and Smart Detection." Microsoft Learn, 2024.
- [16] National Institute of Standards and Technology (NIST). "AI Risk Management Framework (AI RMF 1.0)." U.S. Department of Commerce, 2023.
- [17] OpenAI. "Function Calling and Tool Use in Large Language Models." OpenAI Platform Documentation, 2024.
- [18] PagerDuty, Inc. "State of Digital Operations Report." PagerDuty Research, 2024.
- [19] SAP SE. "SAP Focused Run: AI-Based Anomaly Detection for SAP Landscapes." SAP Documentation, 2024.
- [20] SAP SE. "SAP Joule Agents: Multi-Agent Orchestration Overview." SAP News Center, 2024.
- [21] Splunk Inc. "Splunk IT Service Intelligence: Predictive Analytics Documentation." Splunk Documentation, 2024.
- [22] The Linux Foundation / CNCF. "OpenTelemetry Specification." CNCF Documentation, 2024.
- [23] U.S. Cybersecurity and Infrastructure Security Agency (CISA). "Secure by Design Principles." CISA Publications, 2023.
- [24] World Economic Forum. "Global Risks Report 2025: Technology and Infrastructure Resilience." WEF Publications, 2025.
- [25] Wu, J. "Kubernetes Site Reliability Engineering: Automated Remediation Patterns." O'Reilly Media, 2023.

Appendix A: Glossary of Terms

Autonomy Ladder - A graduated model (L0 through L5 in this article) describing how much independent decision-making and action authority a self-healing system is granted.

Blast Radius - The scope of systems or services potentially affected by an incident or by a remediation action taken in response to it.

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.

GenAI (Generative AI) - AI systems, typically built on large language models, capable of generating novel text, reasoning, or plans in response to a prompt or context.

MTTR (Mean Time to Resolve) - The average time elapsed between an incident's detection and its full resolution.

Retrieval-Augmented Generation (RAG) - An architecture pattern in which a generative AI model's output is grounded by retrieving relevant context at inference time, rather than relying solely on the model's trained parameters.

Root Cause Analysis (RCA) - The process of identifying the underlying cause of an incident, as distinct from its surface symptoms.

Self-Healing Infrastructure - Infrastructure capable of autonomously detecting, diagnosing, and remediating certain classes of operational issues without requiring human intervention for every step.



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