# BIGS AI GOVERNANCE FRAMEWORK

A Decision Framework for Safely Managing
AI Autonomy in Enterprise IT Operations

**Whitepaper • Version 1.0 • July 2026**

**Bigs Bilişim Ltd.**
www.bigsbilisim.com

# 1. Executive Summary

AI agents are rapidly entering enterprise IT operations. Gartner predicts that by the end of 2026, 40% of enterprise applications will feature task-specific AI agents, up from just 5% in 2025. Yet most organizations have no measurable answer to a fundamental question: which operations can an AI agent execute on its own, and which require human approval?

Leaving this question unanswered pushes organizations toward one of two failure modes: uncontrolled autonomy (AI performing unpredictable interventions in production environments) or zero autonomy (AI reduced to an expensive dashboard that never pays back the investment).

The Bigs AI Governance Framework is a production-tested decision framework built to resolve this dilemma. At its core is the **Runbook Confidence Index (RCI)**: a measurable trust model that scores every operational action across five dimensions and maps it to one of four clearly defined autonomy levels. The framework further defines operational safety mechanisms (the 15-Minute Recovery Rule, AI-Assisted Change Risk Scoring, a complete audit trail), an Operational Memory Layer, and a runbook lifecycle in which autonomy is earned over time.

This document presents the framework's principles, its scoring model, its safety mechanisms, and a technology-agnostic reference architecture, followed by a four-phase roadmap for adopting the framework in your own environment.

# 2. The Problem: Between Uncontrolled Autonomy and Zero Autonomy

IT operations teams have lived in the same loop for years: an alert fires, an engineer wakes up, diagnoses the issue, and — more often than not — repeats a standard remediation that has been applied dozens of times before. Industry research indicates that 40–60% of IT triage and repetitive remediations are automatable as of 2026. Large language models and agent architectures make this automation technically possible.

The gap between technically possible and operationally safe, however, is governance. In the field, we observe two typical failure modes:

- **Uncontrolled autonomy:** The AI is granted broad privileges; the system triggers a high-blast-radius operation (say, a database failover) at 3:00 AM based on a false-positive alert. A single bad experience sets an organization's trust in AI automation back by years.
- **Zero autonomy:** Out of fear of risk, the AI only produces 'suggestions' and every step requires a human. Engineer workload doesn't shrink, response times don't improve, and the AI investment turns into a glorified chat interface.

Both modes stem from the same missing piece: a **measurable trust model**. The question 'how much do we trust this operation?' must be answered with a score, not a gut feeling. That is precisely what the Bigs AI Governance Framework does.

# 3. Core Principles

The framework rests on five principles. Every mechanism that follows is an application of these principles:

- **Principle 1 — Autonomy is earned, never assumed.** No action runs autonomously because it is 'probably safe.' Autonomy is the outcome of measurable criteria and historical evidence.
- **Principle 2 — Every autonomous action must be reversible.** Operations that are irreversible, or whose rollback path is uncertain, can never be promoted to full autonomy — regardless of their success history.
- **Principle 3 — An AI without memory cannot be trusted.** The AI must not evaluate every incident from scratch; it must receive past incidents, root causes, and remediation outcomes as context.
- **Principle 4 — A human is always in the loop.** Autonomy does not mean removing humans; it means engaging them at the right moment with the right information. Escalation paths are predefined for every action.
- **Principle 5 — Every decision is traceable.** What the AI did matters — and so does why: the trigger, the score, the decision, the outcome, and the duration are all recorded.

# 4. The Runbook Confidence Index (RCI)

The RCI is the scoring model that determines the autonomy level at which the AI may execute an operational action (a runbook). Each runbook is scored 0–20 on five dimensions, yielding a total score of 0–100.

## 4.1 The Five Dimensions

| Dimension | Question It Asks | High-Score Example |
|---|---|---|
| **Predictability** | How deterministic is the outcome? Does the same input always produce the same result? | Service restart |
| **Reversibility** | If the operation goes wrong, how quickly and completely can it be rolled back? | Change with a configuration backup |
| **Blast Radius** | Does the operation affect a single service, a server, or the entire environment? | Single application pool (narrow scope) |
| **Historical Success** | How many times has this runbook executed, and with what success rate? | 100% success over the last 50 runs |
| **Detection Confidence** | How reliable is the alert that triggers the action? Is the false-positive rate low? | Multi-source, corroborated alert |

## 4.2 The Four Autonomy Levels

The total RCI score maps each action to one of four decision levels:

| RCI Score | Level | Behavior |
|---|---|---|
| 85 – 100 | **Autonomous** | The AI executes the action on its own and reports the outcome. Humans are informed, not consulted. |
| 65 – 84 | **AI Recommendation** | The AI prepares and justifies the remediation; a human executes it with a single approval. |
| 40 – 64 | **Approval Required** | A human reviews the AI's diagnosis and proposal in detail, then approves or rejects with a rationale. |
| 0 – 39 | **Manual Only** | The AI takes no action; it provides diagnosis, context, and prior-incident intelligence only. |

## 4.3 Example Scenarios

| Action | RCI | Level | Deciding Factor |
|---|---|---|---|
| IIS application pool restart | 87 | Autonomous | Narrow blast radius, high historical success |
| Freeing disk space by clearing temporary files | 78 | AI Recommendation | Predictable, but deletion is irreversible |
| Windows service restart on a production application server | 58 | Approval Required | Wide blast radius, dependencies involved |

| Action | RCI | Level | Deciding Factor |
|---|---|---|---|
| Production database failover | 34 | Manual Only | Critical impact, high cost of rollback |

Initial scores are set in a workshop with the IT team during onboarding and are then updated continuously, driven by data, throughout the lifecycle (Section 7). The value of the RCI lies not in mathematical precision, but in moving the autonomy debate from intuition to measurement.

# 5. Operational Safety Mechanisms

## 5.1 The 15-Minute Recovery Rule

If a system does not stabilize within 15 minutes of an autonomous or approved intervention starting, the AI halts the intervention and hands the incident over to a human engineer with full context: the actions taken, the effects observed, and the current working hypotheses. This rule structurally eliminates the risk of an AI 'digging the hole deeper' while trying to correct its own mistake.

## 5.2 AI-Assisted Change Risk Scoring

The framework covers not only incident response but planned changes as well. Every change request is assigned an AI-computed risk score based on the systems affected, the change window, the presence of a rollback plan, and the outcomes of similar past changes. High-risk changes are automatically routed to an additional approval layer.

## 5.3 Privilege Limitation and the Controlled Execution Layer

The AI never accesses target systems directly with unrestricted privileges. It operates through a controlled execution layer: predefined command sets (allowlists), a distinct identity and privilege scope per action, environment separation (production vs. test), and time-windowed access. The set of things the AI can do is bounded by the set of things it is approved to do.

## 5.4 Audit Trail

For every action, the following is recorded: the triggering alert and its source, the computed RCI score with its dimension breakdown, the decision taken (autonomous / approved / rejected), the commands executed, the observed outcome, and the total time to resolution. These records satisfy compliance requirements and serve as the data source for the lifecycle described in Section 7.

# 6. The Operational Memory Layer

The greatest weakness of large language models in enterprise operations is contextlessness: without knowledge of your environment's history, the model evaluates every incident as if seeing it for the first time. The Operational Memory Layer closes this gap.

The layer stores incident history, root-cause analyses, applied remediations, and their outcomes — both structured and as vector embeddings. When a new incident arrives, the AI semantically retrieves similar past incidents and includes them in its decision context: 'A similar memory alert has occurred 4 times on this server in the last 6 months; in 3 cases the root cause was service X, and remediation Y resolved it.'

The memory layer is also the data source feeding the Historical Success and Detection Confidence dimensions of the RCI — without memory, the trust model itself remains static. This is the direct application of Principle 3.

## 7. The Runbook Lifecycle: How Autonomy Is Earned

What turns the framework from a static policy document into a living system is that autonomy levels change over time — and with data:

- **Start:** Every newly defined runbook starts at 'Approval Required' at most, based on its initial RCI score. No runbook is born 'Autonomous.'
- **Promotion:** Every successful execution increases the Historical Success score. When defined thresholds are crossed, the system generates a promotion proposal; the promotion decision is always made by a human.
- **Demotion:** A failed execution or a 15-Minute Rule violation automatically demotes the runbook one level and keeps it there until a root-cause analysis is completed.
- **Periodic review:** The entire runbook portfolio is reviewed quarterly; environmental changes (new dependencies, architectural shifts) are reflected in the scores.

This cycle also builds organizational trust in AI autonomy incrementally: the team sees, in concrete data, that autonomy is earned rather than arbitrary.

## 8. Reference Architecture

The framework is technology-agnostic; the layers below can be realized with different products. The examples given are Bigs Bilişim's choices, validated in production environments:

| Layer | Responsibility | Example Technology |
|---|---|---|
| **1. Monitoring & Detection** | Metric, log, and event collection; alert generation | Zabbix, custom agents |
| **2. Orchestration** | Incident flow management, integrations, approval workflows | n8n |
| **3. AI Decision Layer** | Diagnosis, RCI evaluation, remediation plan generation | LLM (API or local) |
| **4. Operational Memory** | Vectorized storage of incident history, root causes, and outcomes | PostgreSQL + pgvector |
| **5. Controlled Execution** | Privilege-limited, auditable command execution | SSH / PowerShell |
| **6. Visibility** | Real-time status, audit trail, and management reporting | Grafana |

A defining property of the architecture is that every component can run on infrastructure the customer controls — on-premises or in the customer's own cloud. Operational data and system access never have to be handed over to a third-party SaaS platform. For organizations with data sovereignty requirements — GDPR, KVKK, or sector-specific regulation — this is a decisive design choice.

## 9. Adoption Roadmap

Adapting the framework to an organization proceeds in four phases:

- **Phase 1 — Inventory & Runbook Extraction (2–3 weeks):** Existing alert types, recurring incidents, and de facto remediation practices are documented; the initial runbook portfolio is produced.
- **Phase 2 — RCI Scoring Workshop (1 week):** Together with the IT team, every runbook is scored across the five dimensions; autonomy levels and escalation paths are defined.
- **Phase 3 — Shadow Mode (4–8 weeks):** The AI produces a diagnosis and remediation proposal for every incident but takes no action; humans execute the proposals. This phase is the evidence-gathering period in which Detection Confidence and proposal quality are measured.
- **Phase 4 — Graduated Autonomy:** Runbooks validated by shadow-mode data are promoted under the lifecycle rules — first to 'AI Recommendation,' then, for those that earn it, to 'Autonomous.'

## 10. Conclusion

The differentiating question of the next two years will not be 'do you use AI?' but 'how much authority do you give your AI — and based on what evidence?' The Bigs AI Governance Framework provides a measurable answer: autonomy is earned, every action is reversible and traceable, a human is always in the loop, and trust is built with data rather than intuition.

The framework was distilled from the requirements of real production environments during the development of Bigs Bilişim's own AI-powered monitoring and operations platform, and it will continue to evolve with new field experience.

## About Bigs Bilişim

Bigs Bilişim is an Istanbul-based technology company that combines more than 26 years of enterprise systems engineering experience — spanning Windows, Linux, cloud, and SAP infrastructures — with modern automation and AI technologies. The company delivers AI-powered monitoring and operations services, infrastructure consulting, and automation solutions to enterprise customers in Türkiye and internationally.

Contact and further information: **www.bigsbilisim.com**