The Rationalization Window

Turning OSS/BSS Tool Sprawl Into AI-Native Savings

A Business Case for Tools Rationalization in the AI Era

Shawn Ennis

Founder & CEO, Rapax

shawn@citus.cloud

rapax.app | sales@rapax.app

I. The Tool Sprawl Problem

You know the tools. You signed the contracts five, ten, maybe fifteen years ago. They made sense at the time. Now they don't.

Every telecom operator runs some version of the same inventory: two or three OSS/BSS platforms acquired over successive technology cycles, each one solving a problem that existed when it was purchased. A fault management system from one era. A performance management platform from another. A ticketing integration built by a systems integrator who is no longer on retainer. An element management system the vendor stopped investing in three releases ago.

These tools are not just underperforming. They are actively consuming budget. Every year, the operator writes checks for annual maintenance on software that no longer matches how the network operates. Every year, a systems integrator bills professional services hours to maintain integrations between platforms that were never designed to work together. Every year, one or two people on the engineering team spend a meaningful percentage of their time as the caretakers of a tool the rest of the organization wishes would go away.

The operator knows this. The VP of Engineering knows it. The CFO sees the line items. But the tools persist — not because anyone wants them, but because replacing them has historically meant a multi-year, multi-million-dollar transformation program with its own risks, its own SI costs, and its own disruption. The cure has always seemed as expensive as the disease.

This paper argues that the economics of that calculation have fundamentally changed. AI-native platforms that can absorb the functions of legacy OSS/BSS tools — incrementally, not as a big-bang replacement — make it possible to turn the money you are already spending on tools you do not want into savings that fund their own replacement. The budget is not new. It is trapped inside contracts and operational overhead for systems that are no longer earning their keep.

II. The Trapped Budget

The true cost of a legacy OSS/BSS tool is never just the license fee. It is the sum of five cost layers, each of which compounds over time:

Layer 1: License and Maintenance. Enterprise OSS/BSS platforms carry annual maintenance fees typically ranging from 18% to 22% of the original license value.[1] For a platform originally licensed at $1M, that is $180,000–$220,000 per year in maintenance alone — paid regardless of whether the operator uses 100% of the platform's capabilities or 20%. Many operators report using a fraction of the features they pay to maintain.

Layer 2: Infrastructure. Legacy platforms require dedicated compute, storage, and database infrastructure. Servers age, operating systems go end-of-life, and database licenses add their own maintenance streams. The infrastructure cost of running a single legacy OSS platform can range from $50,000 to $500,000 annually depending on scale — infrastructure that exists solely to host a tool the operator wants to retire.

Layer 3: Systems Integration. This is the cost layer most operators undercount, and it is often the largest. Every integration between an OSS/BSS tool and the rest of the operator's environment — ticketing, provisioning, billing, element management — was built by a systems integrator or internal engineer, and it requires ongoing care.[2] When the upstream platform pushes an upgrade, the integrations must be regression-tested. When the downstream system changes its API, the integrations must be rebuilt. Operators routinely spend $200,000 to $2,000,000 per year on SI retainers and project work to keep legacy integrations functioning.

Layer 4: Exotic Skills. Legacy platforms create skill dependencies that become more expensive over time, not less. The talent pool for platforms built on proprietary rule engines, Perl-based correlation logic, or decade-old Java frameworks shrinks every year. Contractors who can maintain these systems command premium rates precisely because the supply of qualified engineers is declining. When the one internal engineer who understands the platform leaves, the operator faces a choice between an expensive contractor and a risky knowledge gap.

Layer 5: Opportunity Cost. Every dollar spent maintaining a tool you do not want is a dollar not spent on a tool you need. Every engineering hour spent caretaking legacy software is an hour not spent on network modernization, customer experience improvement, or competitive differentiation. This cost never appears on a line item, but it compounds relentlessly.

[1] Analysys Mason, Telecom Software Market Forecast 2024-2028; operator interviews on maintenance cost as percentage of license value

[2] TM Forum, Open Digital Architecture: OSS/BSS integration typically consumes 25-40% of total transformation project budgets

> ***The budget to fund an AI-native replacement is not new money. It is trapped inside maintenance contracts, infrastructure hosting, SI retainers, and skills premiums for tools the operator already wants to eliminate.***

The strategic question for operators is not “can we afford to replace these tools?” It is “can we afford to keep paying for them?”

III. The Integration Tax

If the trapped budget is the visible cost of legacy tools, the integration tax is the invisible one. It is the recurring annual expense of keeping systems connected that were never designed to interoperate — and it is the cost that grows fastest as tool sprawl increases.

The Annual Integration Burden

A typical operator running three to five OSS/BSS platforms maintains dozens of point-to-point integrations between them. Each integration carries its own maintenance burden:

Vendor upgrades break integrations. When Platform A pushes a version update that changes its API, every integration touching Platform A must be tested, potentially refactored, and redeployed. Operators report that vendor upgrades routinely consume two to four weeks of SI or internal engineering time per integration affected — not because the upgrade is complex, but because the downstream dependencies are undocumented or fragile.

Staff turnover creates integration risk. Many integrations were built by engineers or contractors who are no longer available. The integration logic lives in scripts, middleware configurations, or ETL pipelines that are poorly documented and understood by a shrinking number of people. When these integrations break — and they do — the operator must either find someone who understands the original design or rebuild from scratch. Both options are expensive.

Every new integration multiplies the maintenance surface. When an operator adds a new BSS platform or adopts a new ticketing system, the number of integration points does not grow linearly — it grows combinatorially. A five-tool environment with full cross-integration has twenty potential integration paths. Each one is a maintenance liability.

The AI-Native Integration Model

An AI-native platform changes the integration equation structurally. Instead of maintaining point-to-point integrations between N platforms, the operator maintains integrations between N platforms and one. The platform ingests data from network elements, BSS systems, and operational tools through a unified collection layer and exposes functionality through a consistent API and agent architecture.

When a vendor pushes an upgrade, one integration adapter is updated — not twenty. When a new data source is added, a single collection pipeline is configured — not a new web of point-to-point connections. The integration maintenance surface shrinks from $O(N^2)$ to $O(N)$.

This is not a theoretical architecture. It is the operating design of platforms built from the ground up on modern data pipelines, containerized microservices, and event-driven architectures — platforms that treat integration as a core capability, not an afterthought bolted on by a systems integrator.

IV. The Operational Dividend

Tools rationalization delivers direct cost savings through contract elimination and integration reduction. But there is a second category of value that compounds over time: the operational efficiency gained from replacing fragmented, manual workflows with AI-assisted operations.

Reducing Human Effort Per Incident

In a legacy multi-tool environment, every network incident requires the operations engineer to perform a series of manual steps before meaningful troubleshooting can begin: identify the affected device, determine what services traverse it, check for related alarms across multiple platforms, locate the relevant troubleshooting procedure, and assemble enough context to make an informed decision. Industry research shows that employees spend up to 30% of their workday searching for information, with nearly half of those searches failing to find what they need.[3]

In an AI-native environment, these steps are automated or dramatically accelerated:

Operational awareness eliminates the daily status-gathering burden. Instead of managers spending ten or more hours per week walking the floor, joining bridge calls, and pulling reports from multiple platforms, an AI agent synthesizes operational status from across the network and delivers executive-level summaries on demand.

Knowledge retrieval replaces manual procedure searches. When an engineer encounters an unfamiliar alarm or device, they describe the problem in natural language and receive the relevant troubleshooting procedure — sourced from vendor manuals, runbooks, SOPs, and institutional knowledge — without needing to know where the document lives or how it is organized.

Contextual enrichment eliminates manual topology assembly. When an alert fires, the system automatically surfaces the affected device's relationships — connected devices, traversing services, customer impact — so the engineer starts troubleshooting with full context instead of spending fifteen minutes assembling it by hand.

[3] Enterprise knowledge management research: employees spend up to 30% of workday searching for information, with nearly half of searches failing

Intelligent notification delivers actionable intelligence instead of raw alarm data. Notifications carry troubleshooting context, affected service information, and relevant procedure references, so the engineer who receives a 3AM page can begin resolving the issue immediately instead of logging into three different platforms to understand what happened.

The Financial Impact of Faster Resolution

The operational efficiencies described above translate directly into financial outcomes. Industry benchmarks consistently place the cost of network downtime at $5,600 per minute or more for enterprise-scale operations.[4] Human error contributes to 66–80% of all downtime incidents, and 85% of those trace back to staff failing to follow procedures or to the procedures themselves being inadequate.[5]

When engineers have faster access to correct procedures, better operational context, and AI-assisted triage, three things happen simultaneously: incidents are diagnosed more accurately (reducing misdiagnosis and rework), resolution actions are executed more quickly (reducing outage duration), and fewer incidents escalate to severity levels that trigger SLA penalties. The compounding effect of these improvements — even at conservative estimates — can represent hundreds of thousands of dollars in annual savings from reduced downtime and avoided SLA credits.

> ***“85% of human-error outages stem from staff failing to follow procedures or from flaws in the procedures themselves.”***
>
> — Uptime Institute, Annual Outage Analysis, 2025

[4] Gartner estimate: average IT downtime costs $5,600/minute. ITIC 2024 survey: 90% of midsize/large companies report 1-hour downtime exceeding $300,000

[5] Uptime Institute, Annual Outage Analysis Report, 2025: 85% of human-error outages stem from procedural failures

V. The Strategic Horizon: The Lights-Out NOC

The first four sections of this paper describe savings that can be modeled, measured, and realized within the first year of deployment. This section describes a strategic outcome that typically takes 12 to 36 months to achieve but represents the most transformative value of AI-native assurance: the transition from a reactive, staff-intensive Network Operations Center to an autonomous, AI-driven operations model.

What “Lights-Out” Actually Means

A lights-out NOC does not mean no humans. It means humans are freed from routine monitoring, triage, and notification tasks to focus on engineering, architecture, and strategic initiatives. AI agents handle the operational baseline: monitoring network health, triaging alerts, correlating events, executing routine remediation procedures, and notifying the right people when human judgment is required. Engineers intervene for novel situations, complex multi-domain issues, and strategic decisions — not for the repetitive work that currently consumes 60–80% of NOC capacity.

This transformation does not happen in a single project. It happens incrementally, as each agent deployed absorbs another category of manual work:

Phase 1 (Months 1–3): Operational visibility and knowledge access. Engineers have real-time status, searchable procedures, and contextual enrichment. Manual effort per incident drops measurably.

Phase 2 (Months 3–6): Notification automation and integration consolidation. Outbound communications are automated. Legacy integration maintenance begins to transfer to the new platform. SI costs start declining.

Phase 3 (Months 6–12): Proactive detection and guided remediation. The platform identifies patterns before they become incidents. Engineers receive guided troubleshooting workflows rather than raw alarm data. MTTR compresses significantly.

Phase 4 (Months 12–36): Autonomous operations baseline. Routine incidents are handled end-to-end by AI agents with human oversight. Engineering staff focuses on network evolution, capacity planning, and customer experience — the work that drives competitive advantage.

Gartner projects that by 2029, 70% of enterprises will deploy agentic AI as part of IT infrastructure operations, up from less than 5% in 2025.[6] Forrester’s 2026 predictions for technology infrastructure and operations identify autonomous AI preventing outages as a critical trend.[7] The operators who begin this transition today build a compounding advantage over those who wait.

[6] Gartner, Predicts 2026: AI Agents Will Transform IT Infrastructure and Operations, December 2025
[7] Forrester, Predictions 2026: Technology Infrastructure and Operations, November 2025

VI. The Savings Model: Three Scenarios

The following scenarios illustrate the financial impact of tools rationalization at three different levels of scope. All current-cost figures are based on representative ranges observed across telecom operators of varying scale. All Rapax figures reflect actual pricing for the corresponding deployment scope.

Scenario A: Single Tool Replacement

Replacing one legacy fault management / correlation / ticketing integration platform

Line Item	Current Annual	With Rapax
Platform maintenance (annual)	$250,000	—
Infrastructure (servers, DB, storage)	$125,000	$15,000
SI / professional services	$500,000	—
Rapax subscription	—	$350,000
White glove onboarding (Year 1 only)	—	$100,000
Year 1 Total	**$875,000**	**$465,000**
Year 1 Savings		**$410,000**
Year 2+ Annual Savings		**$510,000**

In this scenario, the operator eliminates a single legacy platform and its associated integration overhead. The Rapax subscription replaces the platform's functionality — fault management, event correlation, and ticketing integration — at a lower total cost of ownership. Year 1 includes a one-time white glove onboarding investment that covers training, data migration, and integration configuration. From Year 2 onward, the operator saves $510,000 annually — a 58% reduction in total cost for the same functional coverage.

Three-year savings: $1,430,000

Scenario B: Full Assurance Replacement

Replacing the complete service assurance stack — fault, performance, correlation, analytics, and associated integrations

Line Item	Current Annual	With Rapax
Fault management platform (license + maintenance)	$800,000	—
Performance management (license + maintenance)	$500,000	—
Correlation / analytics tools	$400,000	—
Infrastructure (servers, DB, storage)	$600,000	$75,000
SI / professional services	$1,500,000	—
Internal support headcount (2 FTEs)	$300,000	—
Rapax subscription	—	$1,200,000
White glove onboarding (Year 1 only)	—	$250,000
Year 1 Total	**$4,100,000**	**$1,525,000**
Year 1 Savings		**$2,575,000**
Year 2+ Annual Savings		**$2,825,000**

Scenario B represents a mid-market operator replacing their entire service assurance stack. The current environment includes separate platforms for fault management, performance management, and event correlation — each with its own license, maintenance, infrastructure, and integration overhead. Two full-time engineers spend the majority of their time maintaining these platforms and their integrations rather than performing network engineering.

The Rapax deployment consolidates all assurance functions onto a single AI-native platform. The 2 FTEs previously dedicated to platform maintenance are freed to perform network engineering, capacity planning, or other high-value work. SI costs are eliminated entirely as the integration architecture shifts from point-to-point to platform-native.

Three-year savings: $8,225,000

Scenario C: Full Transformation — Lights-Out NOC

Eliminating ticketing, knowledge management, assurance, EMS, IT monitoring, database monitoring, cloud monitoring, and open-source tool overhead — enabling autonomous operations

Line Item	Current Annual	With Rapax
Assurance platform(s) (license + maintenance)	$1,700,000	—
EMS / NMS	$1,200,000	—
Ticketing / ITSM platform	$600,000	—
Knowledge management tools	$200,000	—
IT / server monitoring	$500,000	—
Database monitoring	$300,000	—
Cloud monitoring	$400,000	—
Open-source tool maintenance (internal)	$800,000	—
Infrastructure (all platforms)	$1,500,000	$100,000
SI / professional services	$2,500,000	—
Internal tool support (3–4 FTEs)	$600,000	—
Rapax subscription	—	$2,500,000
White glove + transformation (Year 1 only)	—	$500,000
Year 1 Total	**$10,300,000**	**$3,100,000**
Year 1 Savings		**$7,200,000**
Year 2+ Annual Savings		**$7,700,000**

Scenario C represents a full operational transformation. The operator consolidates every monitoring, management, and operational tool onto a single AI-native platform and uses the resulting operational efficiency to transition toward an autonomous NOC model. This scenario typically unfolds over 12 to 24 months as tools are migrated incrementally — but the savings begin with the first tool decommissioned.

The financial impact is dramatic: the operator eliminates over $10 million in annual tool sprawl costs and replaces them with a $2.6 million AI-native platform that delivers more functionality, requires less infrastructure, and eliminates the need for dedicated tool-support headcount and SI retainers. Three to four FTEs previously dedicated to maintaining legacy tooling are redeployed to network engineering, automation development, or customer experience initiatives.

Three-year savings: $22,600,000

VII. The Try-and-Buy Model

Rapax is not asking operators to take a leap of faith. The commercial model is designed to eliminate risk for both parties.

How It Works

Step 1: Proof of Concept ($25,000). Rapax deploys in the operator’s environment against an agreed scope of functionality. The operator defines the success criteria: specific tools to be replaced, specific capabilities to be demonstrated, specific integrations to be validated. The POC runs in the live network with real data, not in a lab with synthetic traffic. The operator sees exactly what they are buying before they commit.

Step 2: Delivery Against Agreed Scope. Rapax demonstrates replacement of all functionality agreed to during the POC scoping phase. The platform is configured, integrations are built, knowledge bases are loaded, and the operator’s team is trained. Delivery is not measured by features shipped — it is measured by the operator’s confirmation that the platform performs the agreed functions in their environment.

Step 3: Subscription Begins on Demonstrated Value. The subscription does not start on contract signature. It starts when Rapax demonstrates that the product is providing the value agreed to up front. The $25,000 POC investment credits directly to the first subscription payment. The operator pays only after they have confirmed that the platform delivers.

Why This Model Works

For the operator: certainty. You define what success looks like. You see it running in your network. You confirm it meets your requirements. Then — and only then — does the commercial relationship begin. If the POC does not meet the agreed criteria, you walk away having spent $25,000 on a definitive answer rather than millions on a transformation that might not deliver.

For Rapax: alignment. The commercial incentive is to deliver measurable value, not to close a contract and hope for the best. Every deployment begins with a shared definition of success, and payment follows demonstration. This model attracts operators who have a real problem and a real intention to solve it — and it filters out evaluations that were never going to result in a deployment.

The result is a commercial structure that lowers risk for all parties and incentivizes value for money paid. Operators get certainty that they receive what they pay for. Rapax gets a contract guaranteeing payment on delivery of agreed functionality. Both sides are aligned on outcomes from day one.

VIII. Rapax: Built to Replace, Not to Add

Rapax is an AI-native network service assurance platform built by a 25-year telecom operator who has been on both sides of the OSS/BSS procurement conversation. The platform is designed from the ground up to absorb the functions of legacy tools incrementally — not as a multi-year big-bang replacement, but as a phased migration that delivers savings with every tool decommissioned.

The platform's multi-agent architecture means operators do not buy capabilities they do not need. Each agent addresses a specific operational domain:

Oscar (Operations Situational Command Awareness Reporter) delivers operational status summaries, alert forensics, device performance, and quantity reporting — replacing the daily manual status-gathering process that consumes management time across the organization.

Wade (Wisdom Assisted Documentation Engine) provides RAG-enhanced knowledge retrieval across vendor manuals, runbooks, SOPs, and institutional documentation — replacing standalone knowledge management tools, wiki-based SOPs, and the tribal knowledge that walks out the door when experienced engineers leave.

Nora (Network Operations Resource Architect) manages device and service inventory with full provisioning lifecycle support — replacing standalone inventory and provisioning tools and the manual reconciliation processes that keep them synchronized.

Bruce (Business Rules Upgrade Customization Engineer) builds and manages network integrations through guided, AI-assisted workflows — replacing the SI-dependent, point-to-point integration model that generates the integration tax described in this paper.

Grace (Global Response Alert Communications Engine) manages multi-channel notifications with AI-enriched context — replacing standalone notification engines, manual escalation processes, and the communication overhead that delays incident resolution.

Additional agents addressing incident orchestration, automated remediation, compliance validation, revenue integrity, and energy optimization are in active development and extend the platform's rationalization potential over time. Every agent deployed absorbs another category of manual work and another legacy tool's function — compounding the savings with each phase of adoption.

Start the Conversation

You already know which tools you want to eliminate. You already know what they cost. The question is whether the replacement math works.

Let’s find out. A 15-minute conversation is enough to map your current tool landscape, identify the highest-value rationalization targets, and determine whether a $25,000 proof of concept makes sense for your operation. No prep needed. No slide deck. Just a conversation between two people who have lived through enough OSS/BSS procurement cycles to know what matters and what doesn’t.

Book a call: cal.com/shawn-ennis

Email: sales@rapax.app

Learn more: rapax.app

About the Author

Shawn Ennis is the Founder and CEO of Rapax, an AI-native network service assurance platform built for the operators who keep networks running. He holds 12 patents in network management and service assurance, previously founded Assurance companies, and has spent 25 years in telecom operations — including building and selling the kinds of OSS platforms this paper argues should be replaced.