

Infrastructure Intelligence Agent

AI-Powered Operations Co-Pilot for Cloud Infrastructure

Marcin Wypyszynski marcin.wypyszynski@gmail.com

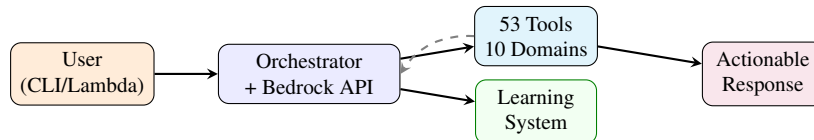
Cloud & Navigation Platform Engineering | April 2026 | Full paper at wypyszynski.dev

Problem

Managing cloud infrastructure requires constant context-switching between ECS, CloudWatch, Cost Explorer, Logs Insights, and Terraform files. Manual investigation across 4 environments and 2 AWS regions is time-consuming, error-prone, and scales poorly.

Solution: AI Agent with Natural Language Interface

An intelligent agent that **reads your infrastructure, understands it, and advises you in plain English**. Powered by Amazon Nova Pro (AWS Bedrock) with a tool-use agentic loop executing up to 30 iterations per query.



10 Capability Domains — 53 Tools

Domain	Tools	Example Query
FinOps Optimization	5	“Which services are over-provisioned and how much can we save?”
Reliability Monitoring	4	“Run proactive health check for production.”
Drift & Compliance	4	“Check drift for all services in dev.”
Log Intelligence	6	“Any anomalies in logs in pre-production?”
Incident Investigation	3	“Reconstruct the charger incident from last 24h.”
Dependency & Blast Radius	6	“What is the blast radius of this service?”
Network & Queue Visibility	13	“Show queue and VPC risks in production.”
Terraform Productivity	4	“Propose right-sizing Terraform changes.”
Knowledge & Learning	8	“What changed since last week for search-service?”
Executive Reporting	–	“Generate executive health report.”

Adaptive Learning System

The agent **gets smarter over time** without model fine-tuning:

- **Learning Engine** — auto-extracts insights from every tool result (non-blocking)
- **Baseline Tracker** — tracks CPU, memory, cost baselines; detects deviations automatically
- **Feedback Collector** — incorporates user approvals, rejections, and corrections
- **Playbook Store** — saves successful multi-step workflows (≥ 3 tools) as reusable templates

Security by Design

Control	Implementation
Read-only default	All write operations blocked unless explicitly enabled
Approval gates	Terraform apply requires human confirmation
Secret masking	Credentials redacted before LLM processing
IAM least-privilege	Minimum required permissions per service
Full audit trail	All tool calls and decisions logged to DynamoDB
Data residency	All processing in-region via AWS Bedrock

Production Results

Task	Manual	Agent	Reduction
Cross-env cost optimization	2–4 hours	2 min	98%
Drift detection (all envs)	1–2 hours	45 sec	99%
Incident root cause analysis	30–90 min	3 min	95%
Environment parity check	45–60 min	30 sec	99%
Executive health report	1–2 hours	1 min	98%

Scope	Scale	Cost	Errors
4 environments, 2 regions, 9+ services	53 tools, 30 iterations/query	\$7–23/month	0

Technology Stack

Amazon Bedrock (Nova Pro) AWS Lambda (Python 3.12) DynamoDB EventBridge Terraform
Click + Rich

© 2026 Marcin Wypyszynski. Full paper with architecture details, learning system design, security model, and production evaluation at wypyszynski.dev.