

● CUSTOMER STORY · FORGE MANAGED RUNTIME

Modernization + FinOps. On Forge, on AWS EKS.

A leading software and market intelligence company consolidates CloudBees CI, GitLab, SonarQube, and JFrog onto one FinOps-disciplined Forge Managed Runtime on AWS EKS — workload-separated, cost-controlled, and governance-ready for the AI workloads coming next.

THE CUSTOMER**Software & Market Intelligence**

Marketing Technology · Market Intelligence
AWS EKS-resident DevOps toolchain

AT A GLANCE

- Workload separation across the chain
- Savings Plan commitments preserved
- Zero production disruption

4

Tools on one runtime

0

Production disruption

0

Idle agent cost

100%

Savings Plan preserved

✓ THE RESULT

Workload-separated, cost-controlled DevOps on AWS EKS — every CI workload class (controllers, agents, analysis, artifacts) in its own pool, scaled and governed on its own terms.

THE ENGAGEMENT

Workloads separated. Cost controlled. Audit ready.

CHALLENGE

One pool, mixed workloads

CloudBees controllers, build agents, SonarQube analysis, and JFrog Artifactory all sharing node capacity, scheduling logic, and FinOps fate. Cost analysis surfaced over-provisioning; peak loads surfaced instability.

SOLUTION

Workload-separated on one EKS

Structured modernization within the existing AWS EKS environment — Kubernetes labels, taints, and tolerations route every workload class to the right pool, with no instance-type or Savings Plan changes.

RESULTS

Cost down, separation up

Agents and analysis scale to zero. Savings Plans intact. Zero production disruption. Evidence streams ready for Reign — GitLab, SonarQube, JFrog, CloudBees all emitting audit-grade records.

“

Forge gave us one runtime, one SLA, and one audit trail across CloudBees, GitLab, SonarQube, and JFrog — modernized, FinOps-disciplined, and governance-ready.

— Director of Platform Engineering (paraphrase, pending customer approval)

THE TOOLCHAIN BEFORE

Despite running a mature CloudBees CI environment on AWS EKS, the company's broader DevOps toolchain — controllers and agents, GitLab runners, SonarQube analysis pods, JFrog Artifactory — was sharing node capacity, scheduling logic, and FinOps fate.

WHAT THE COMPANY NEEDED

Not a platform replacement. A structured modernization within the runtime they already trusted — one that would let each workload class scale and be governed on its own terms, with FinOps discipline applied uniformly.

HOW ITMETHODS ENGAGED

Engagement began upstream — utilization review across all four tools, then staged modernization within the existing AWS EKS environment with no changes to instance types or Savings Plan commitments. A Forward Deployed Engineer owned design, staging validation, and production cutover.

THREE MOTIONS ACROSS THE TOOLCHAIN

Modernization. Optimization. Governance-ready.

01 · MODERNIZATION

One managed runtime on AWS EKS

CloudBees CI, GitLab, SonarQube, and JFrog Artifactory + Xray consolidated under Forge Managed Runtime on AWS EKS. Kubernetes labels, taints, and tolerations route every workload class to the right pool automatically.

DELIVERED

- ✓ CloudBees on isolated controller pool
- ✓ Build agents — scale-to-zero pool
- ✓ GitLab runners — dedicated pool
- ✓ SonarQube analysis — on-demand pool
- ✓ JFrog Artifactory — stable pool

02 · OPTIMIZATION

FinOps as a property of the runtime

Workload separation identified through structured AWS architecture review. No changes to existing instance types, ARM64 node group, or Savings Plan commitments. Staged cutover with zero production disruption.

OUTCOMES

- ✓ Workload separation across the chain
- ✓ Agents + analysis scale to zero
- ✓ Savings Plan commitments intact
- ✓ Zero production disruption

03 · GOVERNANCE-READY

Evidence streams ready for Reign

The modernized toolchain emits structured evidence — GitLab MR audit, SonarQube quality gates, JFrog Xray vulnerability + SBOM, CloudBees release records — ingest-ready for Reign's Evidence Engine when deployed.

EVIDENCE STREAMS

- ✓ Change-management — GitLab MR audit
- ✓ Code quality — SonarQube gates
- ✓ Vulnerability + SBOM — JFrog Xray
- ✓ Release evidence — CloudBees
- ✓ All ingest-ready for Reign

NEXT STEPS

Modernize. Optimize. On Forge.

Take the next step toward governed AI. Every day without visibility is another day of unaudited AI actions and blind spend.

01

Assess

Architecture and FinOps review across CloudBees, GitLab, SonarQube, JFrog on EKS.

02

Modernize

Workload-separate the toolchain on Forge — zero disruption, Savings Plans preserved.

03

Govern

Deploy Reign over the same runtime — runtime policy + audit-grade evidence.

INCLUDED WITH FORGE

- ✓ Full audit trail of every AI agent action
- ✓ MCP connection governance and approval workflows
- ✓ Real-time cost dashboards with token-level visibility
- ✓ 24/7 enterprise support with dedicated TAM

Schedule a demo: itmethods.com/forge

SOC 2 Type II Certified · 100+ Enterprise Customers · 20+ Years Enterprise Trust

© 2026 iTmethods, Inc. All rights reserved.

FORTRESS FAMILY

Forge | Reign | BioCompute.ai