

CI/CD as a Service



Overview

JioCloud's Application CI/CD as a Service brings everything your team needs - version control, builds, testing, artifact storage, and deployment - into one streamlined platform. Your developers push code to Gitea, Jenkins kicks off builds and tests, Harbor handles container images, and everything's deployed securely to Kubernetes or VMs. It's built on proven tools, managed entirely by us - no patching, no firefighting, no infrastructure overhead.

Key Features

- **Built-in version control**
Use Gitea for Git repositories with team-friendly collaboration features.
- **Fully managed build pipelines**
Jenkins takes care of builds and tests - configured, secured, and tuned for scale.
- **Private container registry**
Store, manage, and version container images securely with Harbor.
- **Simple deployment**
Push to Kubernetes or virtual machines with just a few clicks.
- **Access control made easy**
Define fine-grained permissions for repositories, pipelines, and images.
- **Reusable pipelines**
Share templates across teams to enforce quality and speed.
- **Traceability built in**
Track every change from code commit to production.
- **Scales when you need it**
Auto-adjusting infrastructure keeps pipelines fast and responsive.

Benefits

- **Get to production sooner**
Move from idea to deployment in hours, not days.
- **Fewer errors, better releases**
Versioned, repeatable pipelines mean no surprises.
- **Zero Ops overhead**
We take care of updates, scale, and availability.
- **Security from the start**
RBAC, audit logs, and image scanning come baked in.
- **Self-service, with built-in governance**
Let developers self-serve while platform teams maintain control.

Technologies Supported

- **SCM and Pipelines:** Gitea, Git, Jenkins (freestyle/declarative), Webhooks
- **Container Images:** OCI/Docker images, Helm charts (via Harbor) signature and provenance
- **Protocols:** HTTPS, SSH, OCI distribution, Git (smart HTTP), Webhook events
- **Environments:** Jio Cloud ROS (OpenStack), Azure HCI

Technical Specifications

Availability and Scale

- Control plane: 3 node HA (SCM + Jenkins master + Harbor core HA)
- Runners: autoscaled Kubernetes pods with configurable concurrency

Performance

- Parallel pipelines; artifact caching; image layer cache reuse
- Built in job queue management; rate limited pulls/pushes

Security

- TLS everywhere, network policies, per tenant isolation
- Image signing and verification; CVE policy gates in pipelines

Data Protection

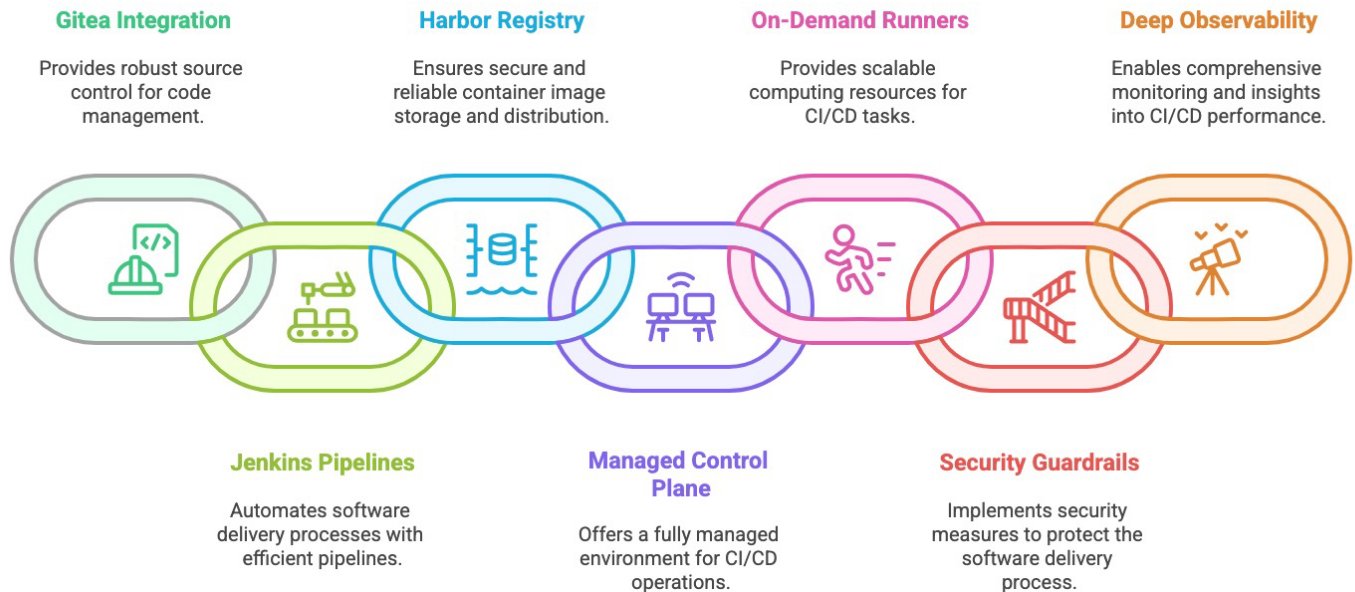
- Scheduled backups of repos, Jenkins state, and Harbor metadata
- Registry GC and retention policies

Observability

- Metrics (pipelines, runners, registry), logs, traces; alerting hooks

Architecture Diagram

JioCloud Managed CI/CD



Data flows (at a glance):

- Code → Gitea → Jenkins pipeline triggers → K8s runners build → Harbor push/sign/scan → deploy.
- Metrics/logs emitted across SCM, pipelines, and registry for dashboards and alerts.

Use Cases

- **End-to-end automation**
Developers push code to Gitea, Jenkins builds and tests it, Harbor stores the image, and Kubernetes handles deployment.
- **Full traceability**
Track artifacts from source commit to production deployment - know what went where, and when.
- **Policy-driven consistency**
Shared pipeline templates enforce code quality checks and security scans across all teams.
- **No infrastructure burden**
Everything runs on infrastructure fully managed by JioCloud - your teams never touch a server.