

Integration Across Multi-Platforms and Elastic Execution within SQAaaS

SQAaaS Engine Architecture and Runtime Environments

Samuel Bernardo

Ibergrid 2026

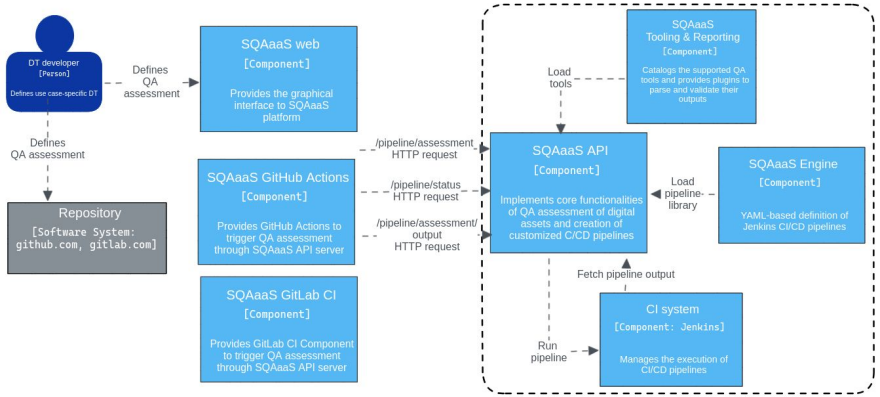
Architecture Overview: SQAaaS Engine & Core Runtime Model

The SQAaaS runtime environment is built around the SQAaaS API and the SQAaaS Engine.

SQAaaS API: Acts as the central interface, receiving HTTP requests to manage pipelines (e.g., `/pipeline/assessment` to create an assessment pipeline, `/pipeline/{pipeline_id}/output` to fetch results). It orchestrates the QA assessment of digital assets.

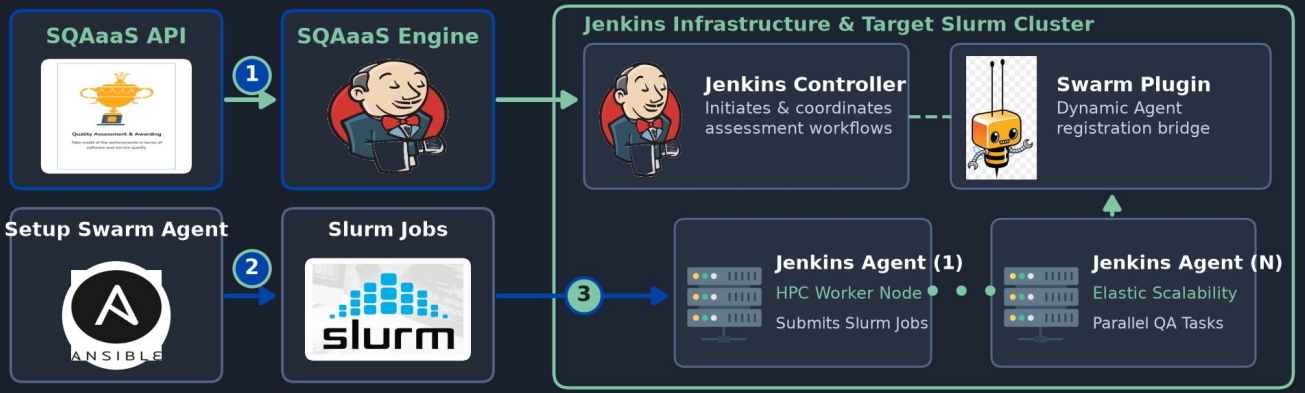
SQAaaS Engine: Handles the YAML-based definition of Jenkins CI/CD pipelines. It generates the necessary pipeline code based on the requested assessment criteria.

Jenkins CI System: The actual execution of the CI/CD pipelines is managed by Jenkins, which pulls the pipeline definition and runs the required tasks.



HPC Execution: SQAaaS Engine using Slurm Clusters

SQAaaS can execute assessments on High-Performance Computing (HPC) environments using Slurm.



1 Integration

The **SQAaaS API** communicates seamlessly with the **SQAaaS Engine**.

The Engine directly interacts with the Jenkins Controller to initiate assessment workflows.

2 Swarm Agent Setup

Utilizes the **Jenkins Swarm Plugin**.

Establishes and configures a dynamic Swarm Agent within the target Slurm cluster environment.

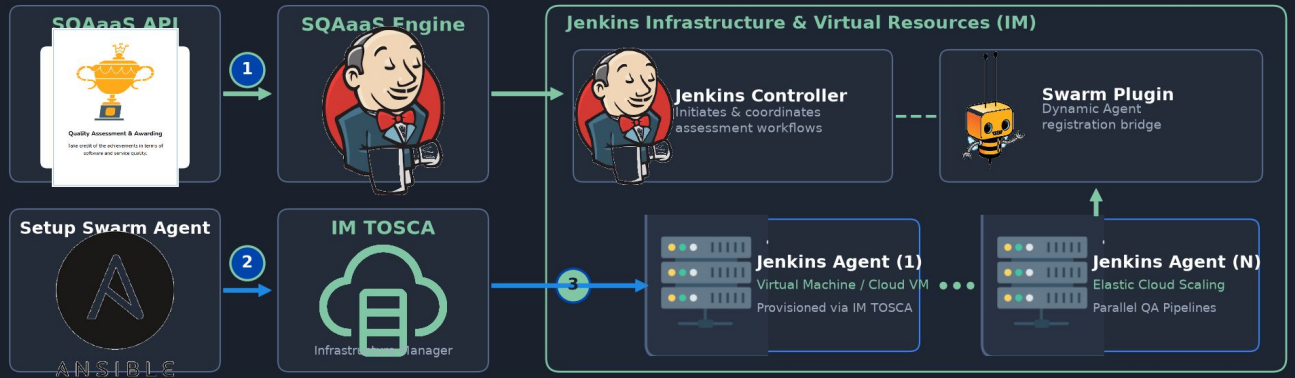
3 Job Execution

The **Jenkins Agent** submits and manages Slurm Jobs.

Executes required Quality Assurance (QA) assessment tasks directly on HPC cluster resources.

Virtualization Execution: SQAaaS Engine with Infrastructure Manager (IM)

Execution on virtualized infrastructure is supported through integration with the Infrastructure Manager (IM).



1 Orchestration

The **SQAaaS API** communicates with the SQAaaS Engine.

The Engine triggers VM-based assessment execution via the Jenkins Controller.

2 IM TOSCA Setup

Utilizes **Infrastructure Manager (IM)** and TOSCA templates.

Configures and provisions target virtual machine infrastructure on demand.

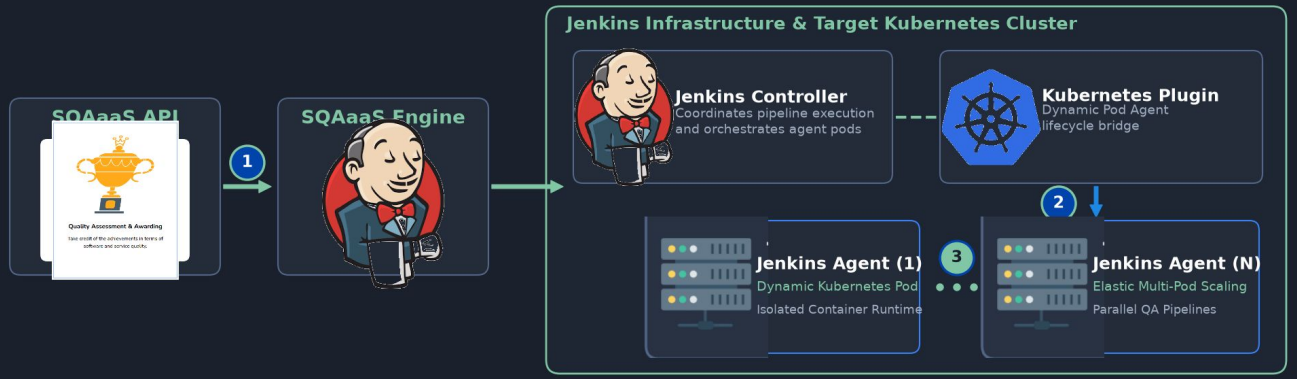
3 Agent Deployment

The provisioned VM runs a **Jenkins Swarm Agent**.

Dynamically registers with the Swarm Plugin to execute QA assessment pipelines.

Container Orchestration: SQAaaS Engine on Kubernetes

Jenkins natively supports elastic execution using Kubernetes for container orchestration.



1 Integration

The **SQAaaS API** triggers assessment workflows via Engine.

Integrates directly with Jenkins Controller to initiate containerized runs.

2 Dynamic Provisioning

Utilizes the **Kubernetes Plugin** for Jenkins.

Dynamically provisions and manages on-demand agent pods inside the cluster.

3 Scalable Execution

Isolated pod agents execute **QA Assessment Pipelines**.

Enables elastic auto-scaling and concurrent multi-pod execution environments.

Configuration as Code: Multi-Platform Runtime Abstraction

SQAaaS Engine abstracts heterogeneous runtimes while preserving strict security isolation outside target platforms

Local & Remote Kubernetes

Jenkins Kubernetes Plugin: Dynamically provisions lightweight Jenkins Agent pods directly on target clusters.

Cross-Cluster Execution: Enables SQAaaS Engine to run assessment jobs across both on-premise local and remote Kubernetes environments on demand.

External Orchestration: SQAaaS service resides securely outside target clusters, orchestrating pipelines via API.

Cloud Virtualization (IM)

Infrastructure Manager (IM): Provisions customized virtual infrastructure across OpenStack, AWS, and Cloud sites using TOSCA templates.

Dynamic Swarm Agents: Spins up on-demand Jenkins Swarm Agents inside provisioned VMs to execute complex assessment suites.

Elastic Lifecycle: Resources are harvested, scaled, and torn down automatically upon job completion.

HPC Clusters (Slurm)

Slurm Batch Orchestration: Directly submits and coordinates high-performance compute jobs on Slurm partition nodes.

Strict Security & Outbound Proxy: Swarm Agents remain completely isolated behind institutional firewalls, connecting via outbound proxy only.

Zero Inbound Exposure: Internal cluster services and data remain fully protected from public Internet exposure.

SQAaaS Engine Agent Selection: Configuration as Code Schema

Declarative YAML schema binding assessment criteria to specialized infrastructure agents

Declarative YAML Schema

```yaml
version: '2.0'
sqaaas_engine:
  environment:
    runtime: kubernetes
    # slurm_hpc | im_cloud
  agent_selection:
    type: dynamic_pod
    pool: k8s-qa-agents
    security:
      outbound_proxy: true
      inbound_rules: blocked
  criteria:
    - step: QC.Sty
    - step: QC.Sec
    - step: QC.Uni
```

Declarative Agent Binding

Declarative Binding: Pipeline assessment criteria (e.g., QC.Sty, QC.Sec, QC.Uni) map directly to target agent profiles defined in YAML.

Agent Specialization: Engine assigns tasks dynamically to specialized agent types: dynamic Kubernetes pods, IM Cloud VMs, or HPC Slurm workers.

Configuration Decoupling: Developers specify standardized QA requirements without needing low-level infrastructure configuration.

Secure Out-of-Platform Brokerage

External Controller Architecture: The SQAaaS API and Engine operate outside target platforms, orchestrating pipelines without cluster privileges.

Outbound Proxy Security: Agents inside protected Slurm HPC or private Cloud VPCs connect outbound only back to the Jenkins Controller.

Zero Ingress Attack Surface: Target computing clusters require no open inbound ports or public IP addresses, maintaining strict security isolation.

Security & Network Isolation: Agent & Controller Infrastructure

Security and network isolation are critical considerations in the SQAaaS architecture.

Isolated Agents

01

Jenkins Agents can be completely isolated behind a firewall. They can use a proxy to establish a connection back to the Jenkins Controller.

Restricted Controller Access

02

The Jenkins Controller itself can have its access restricted and placed behind a proxy.

Internal Resource Access

03

This setup allows the system to access resources that are not directly exposed to the public Internet. The code is pulled from Git, and the pipeline is created and executed on the target infrastructure, building or pulling necessary container images securely.

In conclusion

“All presented work is planned architecture and not yet available in production”

Planned Work: All multi-platform integrations, dynamic harvesting models, and elastic execution pipelines described in this presentation represent planned architectural designs.

Current Availability: These features (Slurm HPC integration, IM TOSCA cloud virtualization, and Kubernetes dynamic pod agents) are not yet available or deployed in the live SQAaaS service.

Roadmap & Future: Active development and validation are underway to progressively roll out and support these elastic execution environments for the community.

Any questions?

Thank you for your attention!

Samuel Bernardo
sqaaas@ibergrid.eu

slides & details