Case Study

Rebuilding the pipeline for performance

Migrating build and test automation from TeamCity on GCVE to GitHub actions on GCP

The Modernization Engineering Company

SONATA SOFTWARE

Summary

The client's Pro Build and regression test automation pipelines were built on TeamCity and GCVE-based infrastructure, which limited reliability, scalability and speed across development environments. Sonata Software led a comprehensive migration of these pipelines to GitHub Actions using GARM runners on Google Cloud Platform (GCP), and moved the underlying TeamCity infrastructure from GCVE to native GCP using Terraform and Chef. The result is a faster, more reliable and more scalable CI/CD foundation, achieved with minimal disruption to developer workflows and lower infrastructure costs.

Customer overview

A leading global technology company that provides cloud-based HCM and WFM software solutions.

Industry	Headquarters	Revenue
Techology	**Massachusetts**	**$650K**

Pressure points

The client's Pro Build and regression test automation pipelines relied heavily on TeamCity and GCVE-based infrastructure. This setup led to unpredictable feedback loops, slower build and test cycles, and constraints on scalability and quality assurance, while deployments in GCVE added further performance bottlenecks and manual effort.

- Capacity and reliability constraints across GCVE and Miami datacenters disrupted feedback loops
- Slower build and test cycles delayed deployments to lower-level environments
- Manual overhead and performance bottlenecks affected deployments in GCVE
- Infrastructure limitations constrained overall velocity and reliability

Solution highlights

Sonata Software migrated all Pro Build pipelines (PR and Main) to GitHub actions using GARM runners hosted on Google Cloud Platform. The team also migrated the TeamCity infrastructure, including projects, build configurations and history, from GCVE to native GCP, using Terraform and Chef to automate provisioning and configuration.

- Migrated all PR and Main Pro Build pipelines to GitHub actions with GARM runners on GCP
- Introduced new workflows within GitHub actions, removing dependency on deployments in GCVE
- Migrated TeamCity infrastructure from GCVE to GCP using Terraform
- Automated TeamCity agent provisioning and configuration using Terraform and Chef
- Integrated existing utilities, including Build Console and PD alerts, into the new CI/CD setup

Results that speak volumes

- Improved reliability and performance across build and test pipelines
- Simplified infrastructure management by removing VMware overhead and moving to GCP-native services
- Greater transparency and collaboration through event-driven, YAML-based GitHub actions workflows
- A scalable, future-ready CI/CD foundation that reduces operational risk
- Minimal disruption to developer workflows throughout the migration