

# G-REPOSITORY & G-DELTA TEST DATA LIFECYCLE MANAGEMENT

## Overview

Data models often evolve during the development and testing process, especially when dealing with 'greenfield' applications. The data architect may need to change the database schema to reflect those changes. With evolving data models and schemas, the QA staff must make sure the test data they are using matches the most current version of the database schema. This becomes a major challenge for larger organizations with distributed teams.

How do you easily make sure that as data tables change, test data scenarios (Scenarios) and test data cases (G-Cases) are in sync without having to manually track and make changes to Test Data Project components.

The solution is a way to automatically detect schema changes in your environment, synchronize them with GenRocket platform, and update all related Test Data Project components.

## Key Features

- Scheduler for database schema change
- Detailed schema change audit reports that can be used to approve or reject schema change
- Auto refactoring of project versions with any change to database schema
- Automated synchronization of Scenarios and Test Data Cases
- License check through a single G-Repository 'server' component instead of by each local machine.
- Runtime statistical data gathering to report through G-Analytics

## Benefits

- Scheduled and automated schema change detection through G-Delta
- Automatically refactor associated project version in your GenRocket cloud environment without manual intervention. This ensures a Project version is always updated with the latest schema
- Automated synchronization of Scenarios and Test Data Cases in the local environment using G-Repository
- Improved security by having license checks through a single G-Repository “server” component
- Local runtime activity statistics reporting to better understand the utilization of the GenRocket platform

## Use case

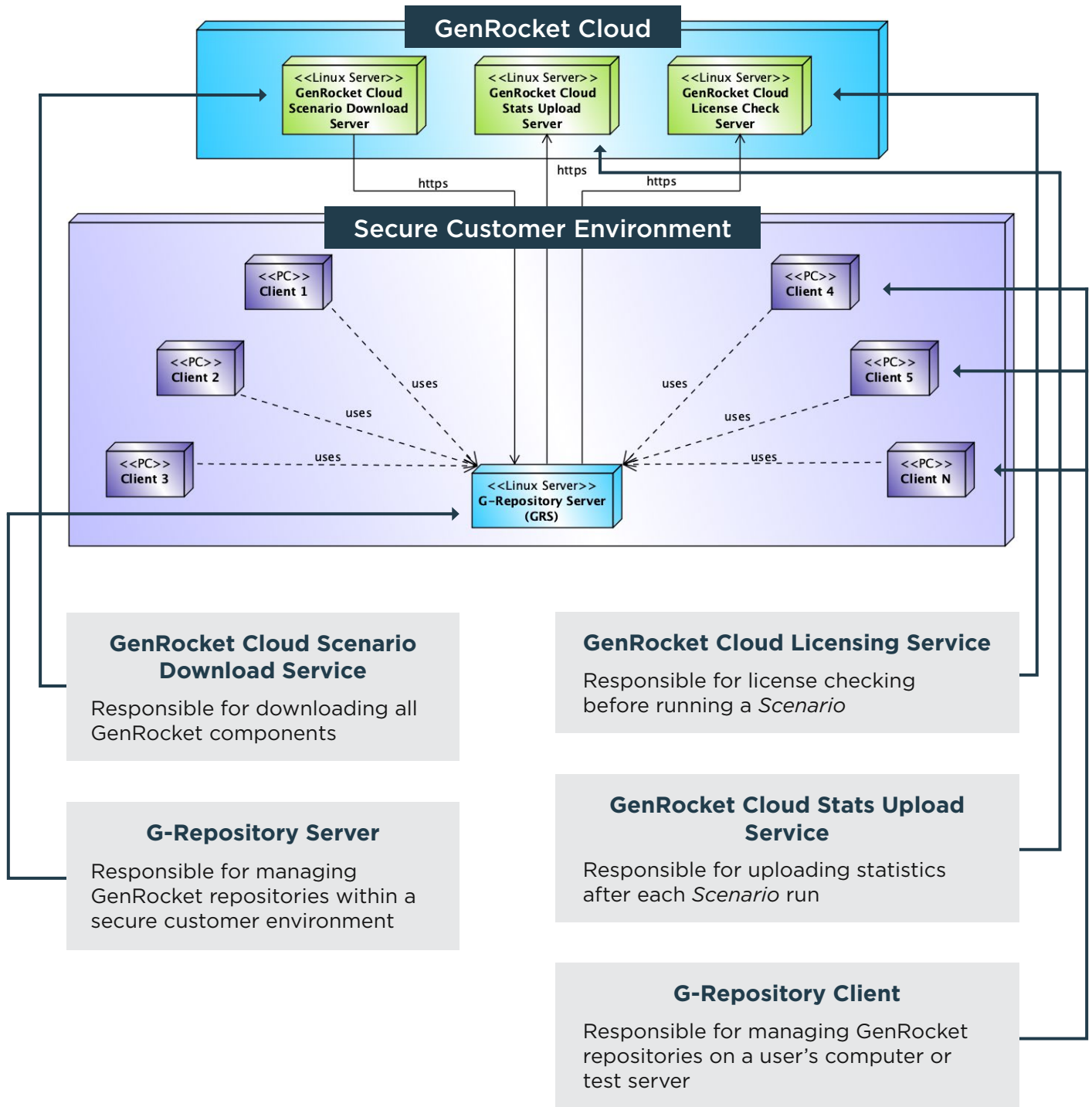
An organization is developing a new application with multiple databases and dynamic schema. There are multiple development and testing teams working on this application. The test data engineers have setup an initial Test Data Project and project version on GenRocket cloud.

During testing, data architects continue to change the database schema by adding/deleting attributes, tables added/deleted, table names changed etc. The problem is with distributed teams, schema changes are not always communicated. Many teams are working on and off shore which makes communication and coordination even more complex.

G-Delta has been configured to conduct multiple schema change detection runs per day. G-Repository has been setup and enabled. Now as schema changes are detected, they are synchronized with the associated Project versions on GenRocket cloud through G-Repository. Those changes are refactored and cascaded down to Scenarios and other Project components.

When a tester runs a G-Case in their local environment, GenRocket will check, through G-Repository, that the latest Scenarios and G-Cases are downloaded. G-Repository will conduct real-time synchronization and run the most recently updated version of Scenarios and G-Cases.

# G-Repository Architecture



## For More Information

Visit these links in the GenRocket Knowledgebase:

[G-Repository Org Admin Overview](#)

[How does G-Repository Server work?](#)

[G-Repository Data Architect Overview](#)