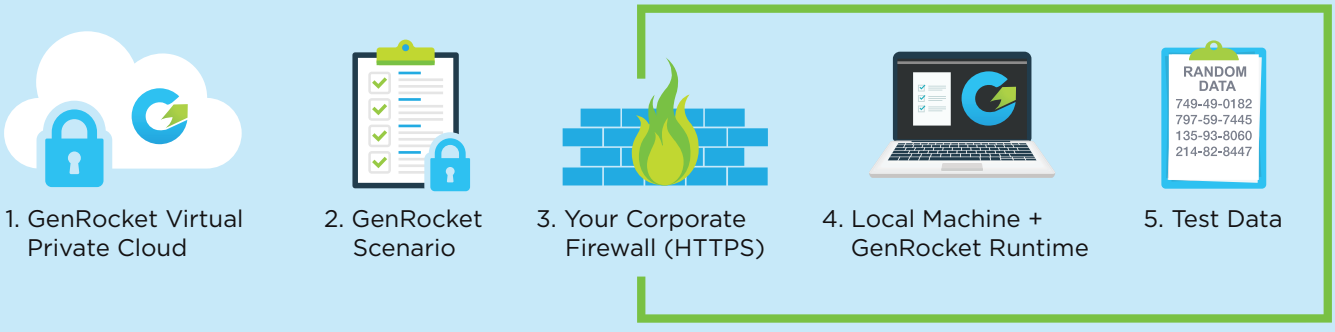
UNDERSTANDING THE GENROCKET ECOSYSTEM

SECURE AND FLEXIBLE TEST DATA AUTOMATION

The GenRocket Ecosystem is a secure *Test Data Automation* platform that adapts to any IT environment. Its component-based architecture makes it easy to define very controlled test data requirements, share projects with co-workers, and run *Test Data Scenarios* that deliver synthetic data blended with production data in real time and during test execution. The GenRocket Ecosystem is designed for seamless integration with test automation tools and allows testers to self-provision any variety or volume of test data on-demand.

The GenRocket Ecosystem



The diagram above provides a high-level view of GenRocket's operating environment. It's a distributed approach that allows teams to work together in the cloud to design controlled and accurate test data scenarios for a given application test plan. After they design the data, it's generated locally and behind the corporate firewall to ensure data security and to maximize system performance.

A MODULAR AND COMPONENT-BASED ECOSYSTEM

The GenRocket Ecosystem is highly modular and utilizes a component-based architecture. It has two major system elements: *GenRocket Cloud* and the *GenRocket Runtime*. Each one plays a different role in GenRocket's *Enterprise Test Data Automation* solution.

GenRocket Cloud: A secure web-based application used by QA teams to model and manage *Test Data Projects* and design test data for different testing Scenarios.

GenRocket Runtime: An on-premise runtime used to execute *Test Data Scenarios* and *Test Data Cases* that generate test data on-demand during test operations.

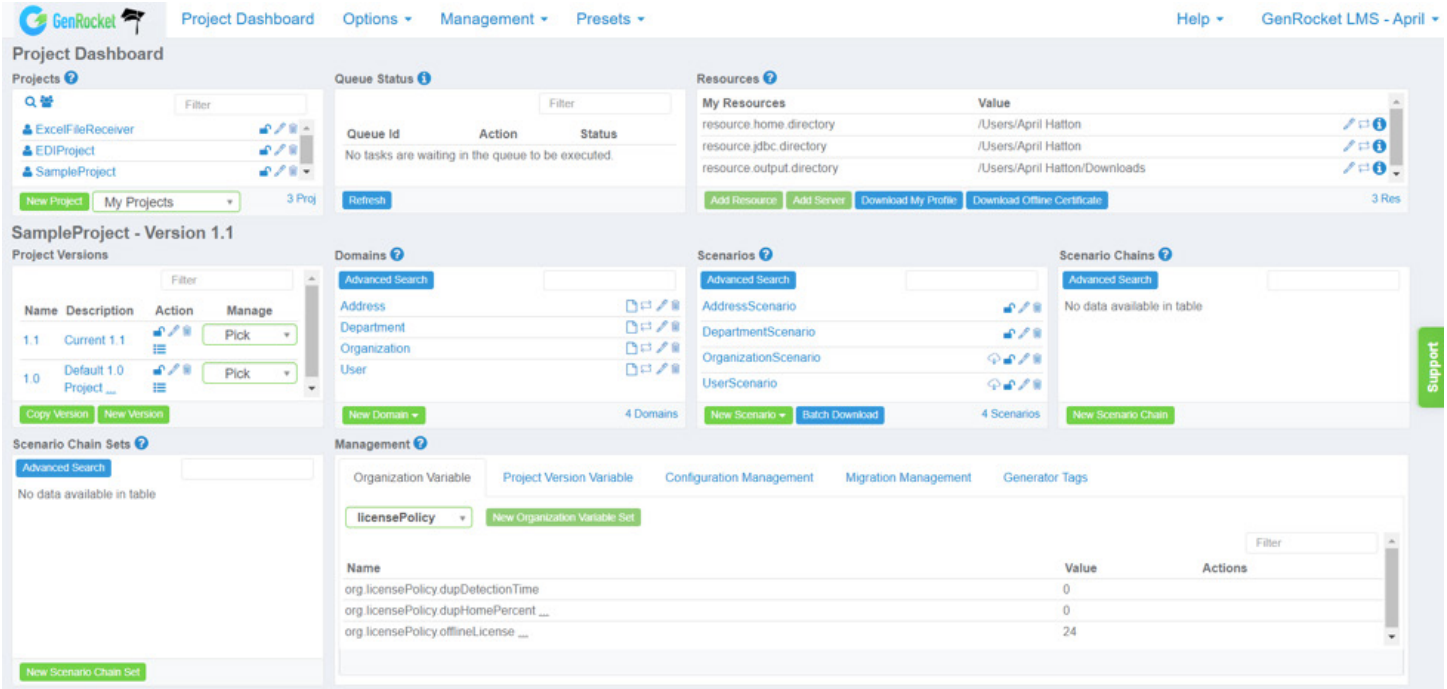
The next section provides a closer look at the components and how they work together to automate the test data provisioning process.

GenRocket Cloud

GenRocket Cloud provides a browser-based graphical interface for users to configure test data based on the five key components used by GenRocket to define any type of test data.

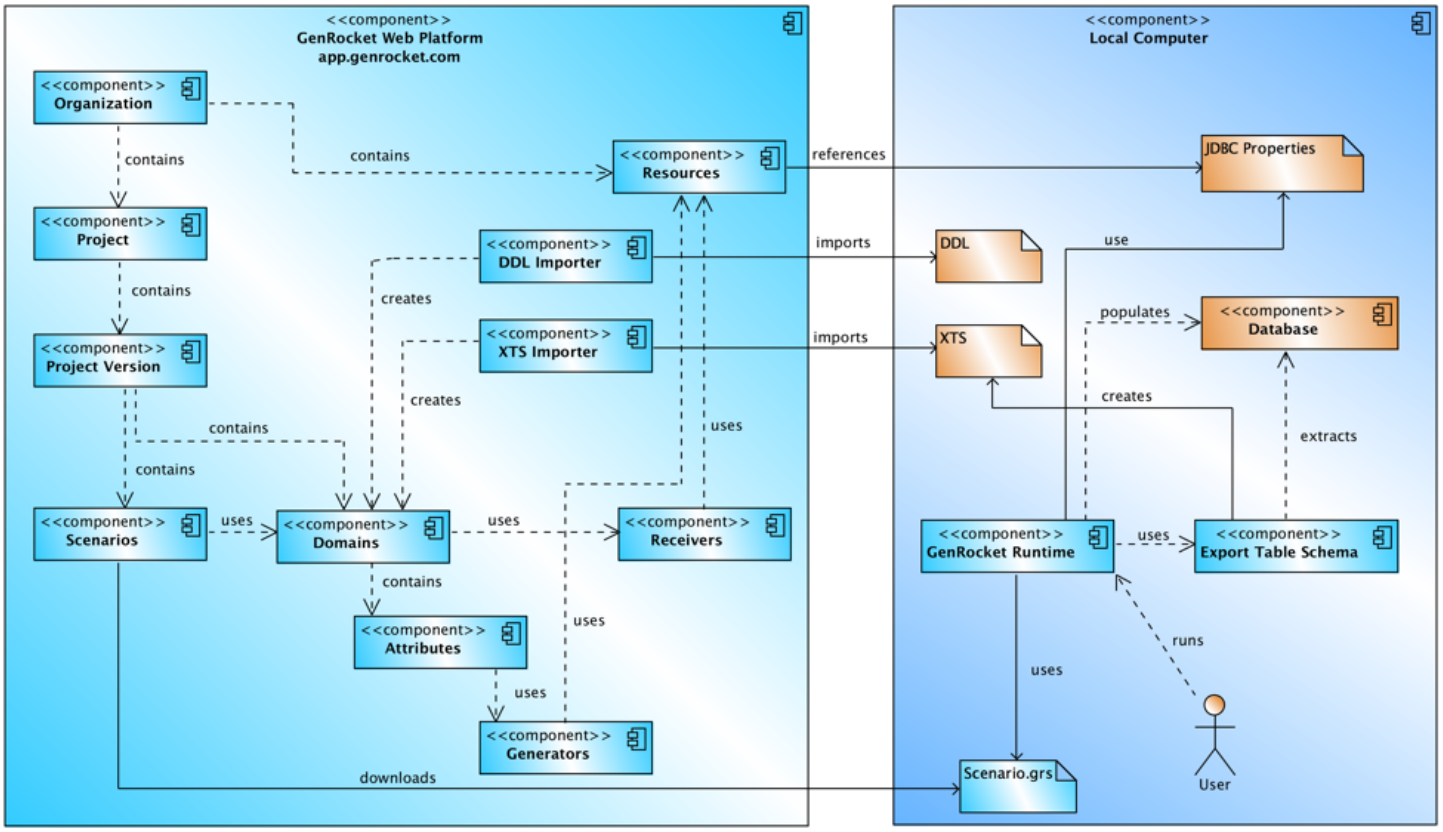
- **Domains:** A person, place or thing. Often defined as a data table in a database
- **Attributes:** The characteristics of the *Domain*. Often defined as the columns in a database
- **Generators:** Generates test data for each *Attribute*
- **Receivers:** Morphs generated data into a useable output format
- **Scenarios:** Instruction sets that tell the *GenRocket Runtime* what data to generate

Many *Scenarios* can be combined in Scenario Chains as part of a test data *Project* that represent test data requirements for an application under test. Here is a screenshot of the *Project Dashboard* in *GenRocket Cloud*. It guides testers through the test data design and configuration process.



Projects can be stored, copied, versioned, and shared across an *Organization*. Users can access all or parts of a Project depending on their permission level. GenRocket offers 9 different methods to quickly set up new Projects; the most popular methods are XTS – Extract Table Schema – where the meta data of an entire database can be used to quickly set up a Project and DDL (Data Definition Language).

GenRocket Cloud is accessed via Chrome, Firefox, and Safari browsers using HTTPS. Single-Sign-On (SSO) is also available upon request. In the diagram below, the left-hand block shows the components used in the *GenRocket Cloud* and their relationships.



The GenRocket Runtime

The *GenRocket Runtime* is illustrated on the right-hand side of the diagram. As you can see in the illustration, the only components that are installed on a tester's machine or test automation server are the GenRocket Runtime and the Scenario files. These Scenario files are the instruction sets that generate test data on demand. As you can see, all data is generated and delivered securely behind the corporate firewall.

GenRocket Runtime Requirements

Although GenRocket is able to run on just about any machine that supports Java, we recommend a minimum system with the following characteristics for basic test data provisioning applications.

- System Hardware: 4 Core CPU with 8GB RAM
- Operating System: Linux, macOS, and Windows
- Java (JDK): Supported releases include the following:
  Release 1.7, 1.9, 1.10. 1.11, 1.12, and 1.13 (Java 1.8 u20 is not supported at this time).

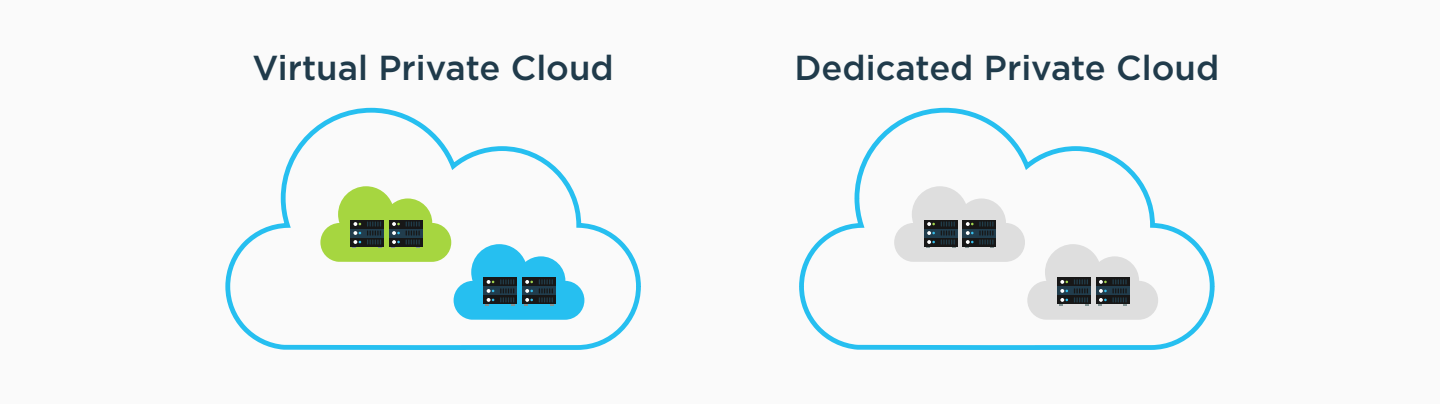
For more advanced test automation environments we recommend your machine meets or exceeds the following specifications:

- 4 Core CPU
- 16 GB RAM
- SSD Hard drive
- Linux/Unix based operating system

View our knowledgebase article covering **installation recommendations and requirements.**

VIRTUAL PRIVATE CLOUD AND DEDICATED PRIVATE CLOUD HOSTING

GenRocket offers Virtual Private Cloud (VPC) and Dedicated Private Cloud (DPC) hosting, but our customers usually use one of two different hosting options for *GenRocket Cloud*. In both cases, the web application is hosted in a secure environment on Amazon Web Services (AWS), a company with world-class facilities, system redundancies and backup procedures.



A *Virtual Private Cloud* is a hosted environment in which multiple tenants cost-effectively share system resources using secure, isolated subnetworks. Most GenRocket customers choose this option. There is no additional cost and the VCP is monitored, managed, and maintained 7 X 24 X 365 by our DevOps team. Software is updated daily on this hosting platform so that customers always have access to the latest features and updates.

A *Dedicated Private Cloud* is an environment in which AWS system resources are dedicated to a single organization. DPC hosting is recommended for large customer projects where there will be more resource-intensive workloads. Like the VPC, the setup for a DPC is monitored, managed, and maintained 7 X 24 X 365 and GenRocket software is updated on a daily basis.

GENROCKET SECURITY CONTROLS

Every aspect of the GenRocket Ecosystem is designed with security in mind. It is important to know that no customer data is stored in the GenRocket Cloud and, as a result, no customer data is ever exposed outside of the corporate firewall.

Each time a user logs in they create a secure, encrypted connection from their browser to GenRocket Cloud. We require a valid username and password and are able to add an extra layer of security through Single Sign-On (SSO). All passwords are encrypted with a SHA-256 one-way hash.

Team Permissions ensures that users of the application gain access to the appropriate features of the application. Four different system roles are available when setting up *Team Permissions* and provide different security access restrictions:

- Organization level (admin)
- Project level
- Domain level
- Scenario level

For more information on GenRocket Teams, see this article: **GenRocket Team Permissions.**

Test Data Scenarios are instruction sets that contain no data and are encrypted before downloading. Only authenticated and licensed users within a given organization can run an encrypted Scenario.

The GenRocket Runtime is a secure Java program that executes encrypted Scenarios and Test Data Cases on-premise, and within the security of the corporate firewall. All GenRocket Engine JAR's are validated with a checksum. For the license check, we call a secure URL and using a secure connection, the check passes back license verification information to GenRocket.

PERFORMANCE AND SCALABILITY

GenRocket's *Test Data Automation* platform is capable of generating thousands of rows of synthetic test data per second. On a given computer, depending on the number of CPU Cores, Memory and Operating System (OS), GenRocket may generate between 10,000 to 15,000 rows of test data per second. If we base our calculations on the idea that GenRocket is running on one very slow computer, then the following test data generation calculations can be approximated:

- 10,000 rows every second
- 600,000 rows per minute
- 1,000,000 rows every 1 minutes and 40 seconds
- 10,000,000 rows every 16 minutes 6 seconds

The GenRocket platform is highly scalable to handle any high-volume test data scenario. The GenRocket Partition Engine is used to generate hundreds of millions, to billions or even trillions of rows of test data in a short period of time. This is accomplished by partitioning the load to generate huge amounts of test data across multiple GenRocket instances running within a given server. **Learn More About GenRocket's Partition Engine**

The GenRocket Ecosystem is a secure, scalable and adaptable platform for Test Data Automation used by some of the largest organizations in the world. Our customers include global banking institutions, financial services companies, healthcare and insurance providers, online retailers and multi-national manufacturing companies. It's a testament to the power of the GenRocket platform, that we have yet to encounter a test data challenge we have not been able to solve.

If you would like to learn more about the GenRocket Ecosystem and how it can be adapted to meet the test data requirements of your organization, schedule a live demonstration of GenRocket Test Data Automation and bring us your test data challenge.

SCHEDULE A LIVE DEMONSTRATION