



A closer look at HP LoadRunner software

Table of contents

Sizing up the system	2
The limits of manual testing	2
A new take on testing: the HP LoadRunner solution	3
The HP LoadRunner solution and terminology	3
Inside HP LoadRunner Vuser technology	4
Protocol-based Vusers	4
The HP LoadRunner testing process	6
Measurable results	8

Sizing up the system

The world is changing. The demand for faster applications has been growing over the years and it is hard for IT to keep up with the quality and performance required by customers.

With new rich Internet applications (RIAs) leveraging Web 2.0 technologies, there are many moving parts, which can easily become points of failure if not tested prior to deployment.

The level of complexity of the applications environment and the new platforms models, as mobile and cloud, add to the challenges of executing performance testing.

For example, a typical application testing scenario might depict 1,000 users that log in simultaneously to a system on Monday morning from different parts of the world. What is the response time of the system? Does the system crash? What is the maximum capacity that the environment will support? To be able to answer these questions, and more, a complete application performance testing solution must do the following:

- Test a solution that combines a variety of software applications and hardware platforms
- Determine the suitability of a server for any given application
- Test the server before the necessary application has been completely developed
- Emulate an environment where multiple clients interact with single or multiple server applications
- Be able to closely simulate the future production environment

Testing performance manually is not an option. A solution to execute application performance testing and minimize the risks of performance failure is fundamental for business success.

HP LoadRunner software is the industry-standard software for performance testing.

Test an application under the load of tens, hundreds, or even thousands of potential users.

HP LoadRunner load tests your application by emulating the environment where multiple users work concurrently. While the application is under load, LoadRunner accurately measures, monitors, and analyzes a system's performance and functionality.

The limits of manual testing

Traditional or manual testing methods offer only a partial solution to load testing. For example, you can test an entire system manually by constructing an environment where many users work simultaneously on the system. Each user works at a single machine and submits input to the system. However, this manual testing method has the following drawbacks:

- It is expensive, requiring large amounts of both personnel and machinery.
- It is complicated, especially coordinating and synchronizing multiple testers.
- It involves a high degree of organization, especially to record and analyze results meaningfully.
- The repeatability of the manual tests is limited.

A new take on performance testing: the HP LoadRunner solution

HP LoadRunner addresses the drawbacks of manual performance testing:

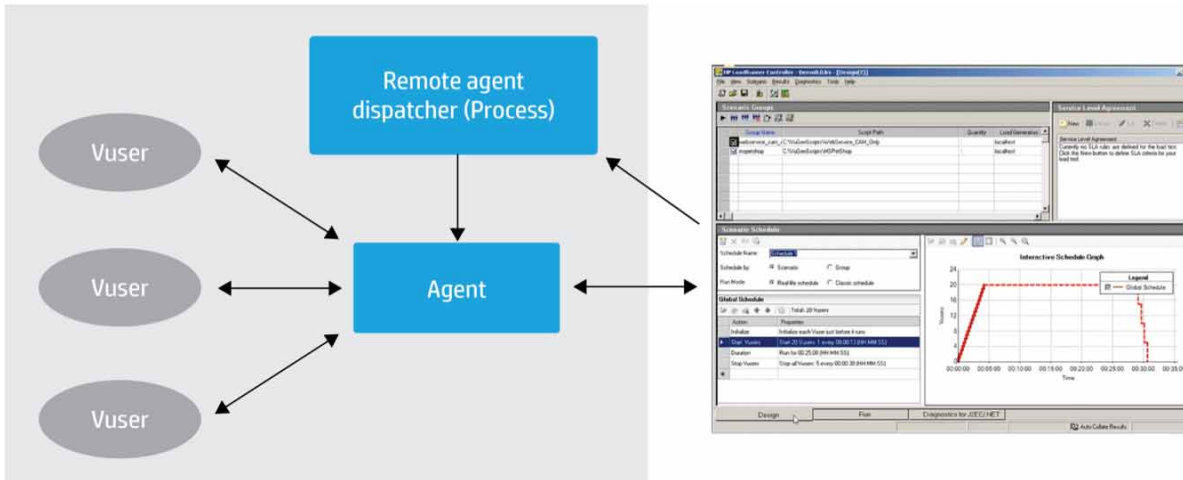
- LoadRunner reduces personnel requirements by replacing human users with virtual users (Vusers). These Vusers emulate the behavior of real users—operating real applications.
- LoadRunner reduces the amount of hardware required for testing because numerous Vusers can run on a single computer.
- The HP LoadRunner Controller allows you to easily and effectively control all the Vusers—from a single point of control or multiple locations to simulate the user's locations.
- HP LoadRunner allows multiple combinations for number of users, locations, timeframes, and other items to simulate the different user behaviors.
- HP LoadRunner provides capabilities to configure and simplify data entry as well as advanced features for data correlation.
- LoadRunner monitors the application performance online, enabling you to fine-tune your system during test execution.
- LoadRunner automatically records the performance of the application during a test. You can choose from a wide variety of graphs and reports to view the performance data.
- LoadRunner checks where performance delays occur: network or client delays, CPU performance, I/O delays, database locking, or other issues at the database server. LoadRunner monitors the network and server resources to help you improve performance.
- LoadRunner tests are fully automated, so you can easily repeat them as often as you need.

The HP LoadRunner solution and performance testing terminology

- **Scenario.** Using HP LoadRunner, you divide your application performance testing requirements into scenarios known as well as user behavior. A scenario defines the events that occur during each testing session. Thus, for example, a scenario defines and controls the number of users to emulate, the actions that they perform, and the machines on which they run their emulations.
- **Vusers.** In the scenario, LoadRunner replaces human users with virtual users or Vusers. When you run a scenario, Vusers emulate the actions of human users working with your application. While a workstation accommodates only a single human user, many Vusers can run concurrently on a single workstation. In fact, a scenario can contain tens, hundreds, or even thousands of Vusers.
- **Vuser scripts.** The set of actions that a Vuser performs during the scenario are described in a Vuser script. When you run a scenario, each Vuser executes a Vuser script. The Vuser scripts include functions that measure and record the performance of your application's components.
- **Transactions.** To measure the performance of the application, you define transactions. A transaction represents an action or a set of actions that you are interested in measuring. You define transactions within your Vuser script by enclosing the appropriate sections of the script with start and end transaction statements. For example, you can define a transaction to process a request to view the balance of an account and for the information to be displayed at the ATM.
- **Rendezvous points.** You insert rendezvous points into Vuser scripts to emulate heavy user load on the server. Rendezvous points instruct Vusers to wait during test execution for multiple Vusers to arrive at a certain point, so that they may simultaneously perform a task. For example, to emulate peak load on the bank server, you can insert a rendezvous point instructing 100 Vusers to deposit cash into their accounts at the same time.
- **Controller.** You use the HP LoadRunner Controller to manage and maintain your scenarios. Using the Controller, you control all the Vusers in a scenario from a single workstation.
- **Load generator.** The load generator is the machine that executes the Vuser script, enabling the Vuser to emulate the actions of a human user. When you execute a scenario, the Controller distributes each Vuser in the scenario to a load generator. The load generator can simulate the different user locations that hits the application.
- **Performance analysis.** Provides a robust view in the performance testing results via reports and graphs. Vuser scripts include functions that measure and record system performance during load-testing sessions. During a scenario run, you can monitor the network and server resources. Following a scenario run, you can view performance analysis data in reports and graphs.

Inside HP LoadRunner Vuser technology

Figure 1: On each Windows load generator, you install the Remote Agent Dispatcher (Process) and a LoadRunner Agent.



- **Remote Agent Dispatcher.** The Remote Agent Dispatcher (Process) enables the Controller to start applications on the load generator.
- **Agent.** The LoadRunner agent enables the Controller and the load generator to communicate with each other. When you run a scenario, the Controller instructs the Remote Agent Dispatcher (Process) to launch the LoadRunner agent. The agent receives instructions from the Controller to initialize, run, pause, and stop Vusers. At the same time, the agent also relays data on the status of the Vusers back to the Controller.

HP LoadRunner supports multiple application platforms

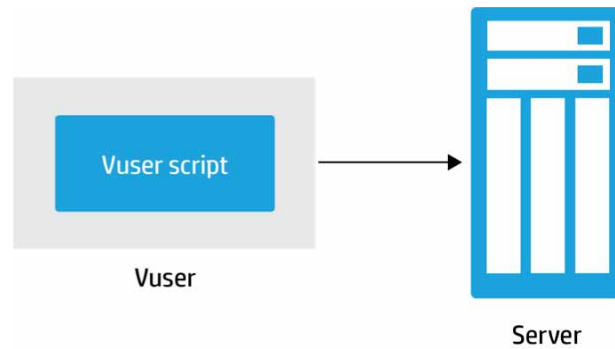
HP LoadRunner supports performance testing for a wide range of application environments and protocols to make scripting faster and easier. Each protocol that simulates the user (Vuser) is designed to handle different aspects of today's system architectures. You can use the protocol types in any combination in a scenario in order to create a comprehensive application test. Protocol types are divided into several categories, indicating their use within common business environments. For example:

- Application deployment solution Vusers support the Citrix or Remote Desktop Protocol (RDP) protocols
- Database and socket communication protocols
- COM/DCOM and Microsoft® .NET protocols
- Oracle NCA, SAP, and Siebel systems
- Jacada, CORBA, RMI, and JMS protocols

Includes also an SDK protocol that allows you to write your own script for specific applications that are not supported by the out-of-the box protocols.

Vusers generate load on a server by submitting input directly to the server. Vusers do not operate client applications—they access the server using LoadRunner API functions. These API functions emulate the input from an actual application.

Figure 2: User load emulation to the application under test (AUT)



Because Vusers are not reliant on client software, you can use Vusers to test server performance even before the client software has been developed. Further, since Vusers do not have a user interface, the amount of system resources required is minimal. This allows you to run large numbers of Vusers on a single workstation.

Performance testing process and HP LoadRunner

Suppose you want to test an online banking Web application that is accessed by many customers via Internet. The website provides a full range of banking services to the customers—such as the ability to transfer funds and check account balances. To test this application, you create a scenario. The scenario defines the actions/behaviors that are performed on the application environment during the load test. During the scenario that loads and monitors the bank application, you want to:

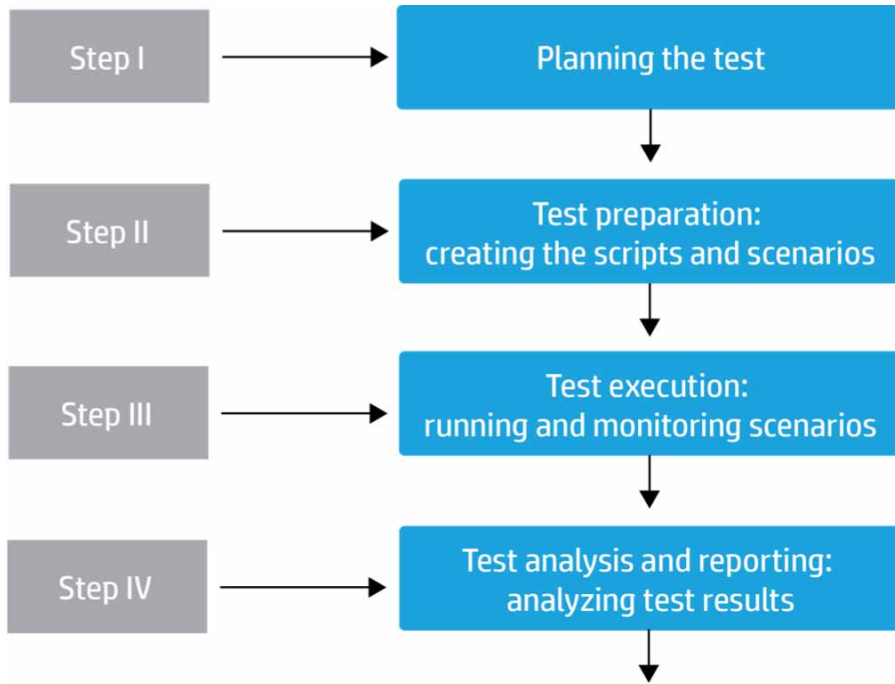
- Emulate conditions of controlled load and maximum load on the application environment
- Measure application performance under load, response time, memory, throughput, etc.
- Check where performance delays occur: network or client delays, CPU performance, I/O delays, database locking, or other issues at the server
- Monitor the network and server resources under load

Once the test is completed, you may want to review the results, perform analysis, and create a report.

Customers can leverage HP LoadRunner for their specific performance projects and requirements since HP LoadRunner is extremely flexible.

Below we provide a high level testing process with HP LoadRunner with four simple steps:

Figure 3: The HP LoadRunner testing process



Step I: Test planning

Successful load testing requires that you develop a thorough test plan. A clearly defined test plan makes sure that the LoadRunner scenarios you develop will help accomplish your load testing objectives.

Planning may include some of the following points:

- Define the team
- Business goals and SLAs
- Application under test
- Critical vs. non-critical transactions within the application under test
- Test environment for application under test including platform, servers, network, database, web services, etc.
- Environment to perform the test
- Test requirements such as response time, load, memory, CPU utilization, etc.
- User behavior, user locations, and application timeframe usability
- Special factors regarding the application under test

Depending on customer requirement, there are other areas of planning for performance testing.

Step II: Test preparation

Vusers emulate human users interacting with your Web-based application. A Vuser script contains the actions that each Vuser performs during scenario execution.

In each Vuser script, you determine the tasks that will be:

- Performed by each Vuser
- Performed simultaneously by multiple Vusers
- Measured as transactions

A scenario describes the events and user behavior that occur that occur during a testing session. A scenario includes a list of machines on which Vusers run, a list of scripts that the Vusers run, and a specified number of Vusers or Vuser groups that run during the scenario. You create scenarios using the Controller.

Creating a manual scenario

You create a scenario by defining Vuser groups to which you assign a quantity of individual Vusers, Vuser scripts, and load generators to run the scripts.

You can also create a scenario using the Percentage Mode, in which you define the total number of Vusers to be used in the scenario, and the load generators and percentage of the total number of Vusers to be assigned to each Vuser script.

Creating a goal-oriented scenario

For Web tests, you can create a goal-oriented scenario, in which you define the goals you want your test to achieve. LoadRunner automatically builds a scenario for you, based on these goals.

Step III: Test execution

You emulate user load on the application by instructing multiple Vusers to perform tasks simultaneously. You can set the level of load by increasing and decreasing the number of Vusers that perform tasks at the same time.

Before you run a scenario, you set the scenario configuration and scheduling. This determines how all the load generators and Vusers behave when you run the scenario.

You can run the entire scenario, groups of Vusers (Vuser groups), or individual Vusers. While a scenario runs, LoadRunner measures and records the transactions that you defined in each Vuser script. You can also monitor your system's performance online.

You can monitor scenario execution using the LoadRunner online run-time, transaction, system resource, Web resource, Web server resource, Web application server resource, database server resource, network delay, streaming media resource, firewall server resource, ERP/CRM server resource, Java performance, J2EE & .NET Diagnostics, application deployment, middleware performance, application component, and infrastructure resources monitors.

HP LoadRunner includes HP SiteScope to enhance the monitor capabilities. At the same time, HP Diagnostics collects application-tier and code-level data to pinpoint the performance issues.

Step IV: Test analysis and reporting

With its Bubble-Up analysis capabilities, HP LoadRunner helps you quickly determine which transactions passed or failed the set service-level objectives, as well as some potential causes of failure. A strong analysis engine helps you slice and dice data in many ways to easily pinpoint the root cause of the problems. You can also use templates to generate multiple custom reports to serve the needs of various stakeholders.

Measurable results

HP LoadRunner software, used by thousands of businesses around the world, is the industry standard software for performance engineering. It enables an efficient and robust means to verifying that your application's architecture is built for more efficient performance and reliability, so you can deploy with high quality and confidence providing business value to your organization.

Generate real life load

Identify and diagnose performance bottlenecks. Deploy with high quality and confidence. Learn more: hp.com/go/loadrunner

Get connected

hp.com/go/getconnected

Current HP driver, support, and security alerts delivered directly to your desktop

© Copyright 2011-2012 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Microsoft and Windows are U.S. registered trademarks of Microsoft Corporation.

4AA3-3960ENW, Created May 2011; Updated June 2012, Rev. 1

