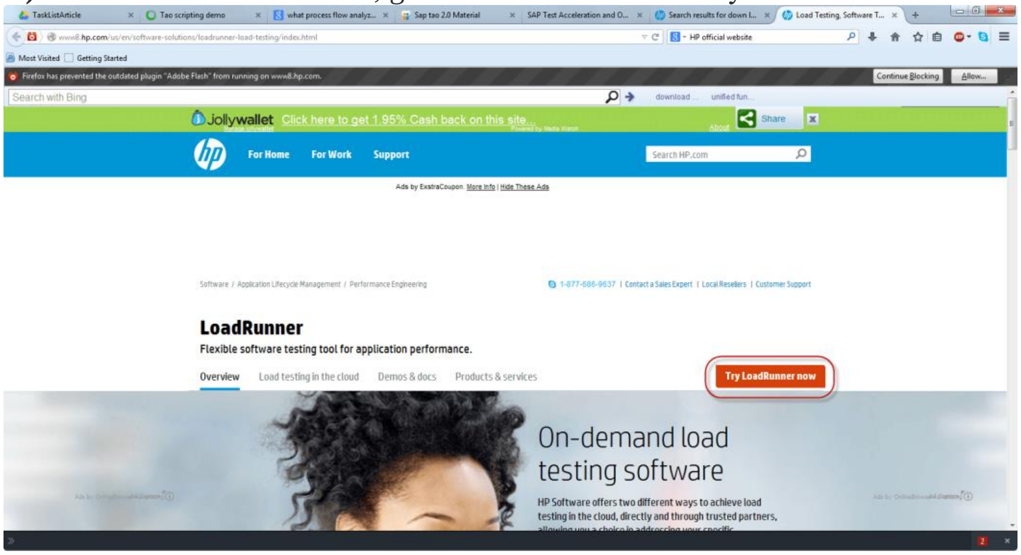
Table of Content

- Download and Install HP LoadRunner 12.0
- Inroduction to Loadrunner and its Architectue
- Understanding VUGen in LoadRunner
- Advance VUGEN: Pamaretization, Transactions, Rendous point
- Correaltion in Loadrunner
- How to use Controller in LoadRunner
- How to use Analysis in LoadRunner

1. **PART 1} Download**

2. **Step 1)** To Download LoadRunner, go to this link and click “Try LoadRunner”

3. 

For Home
For Work
Support
Search HP.com
Software / Application Lifecycle Management / Performance Engineering
1-877-686-9637 | Contact a Sales Expert | Local Resellers | Customer Support
Overview
Load testing in the cloud
Demos & docs
Products & services
Try LoadRunner now

Firefox has prevented the outdated plugin "Adobe Flash" from running on www8.hp.com.

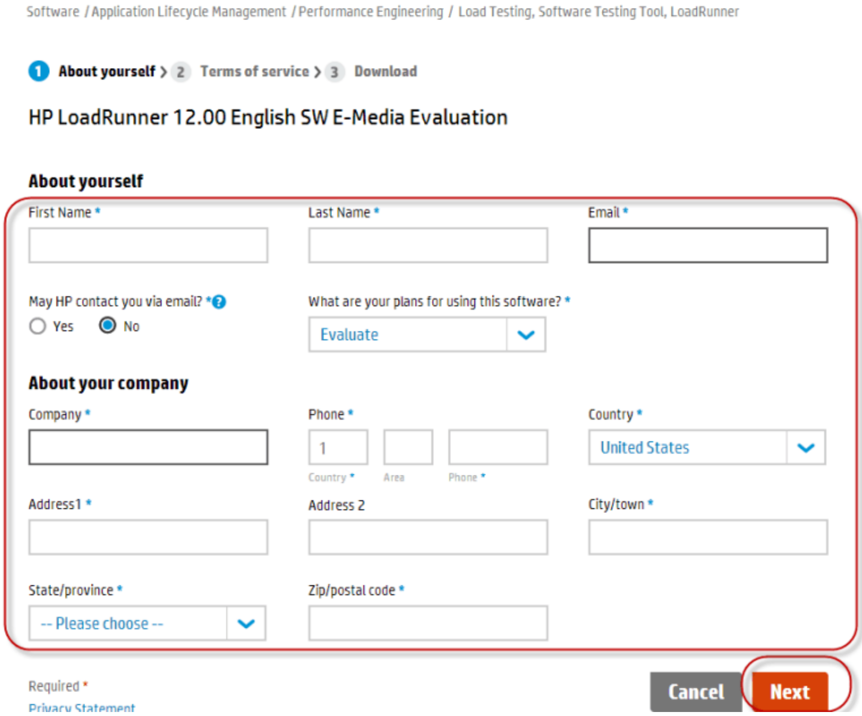
LoadRunner

Flexible software testing tool for application performance.

On-demand load testing software

HP Software offers two different ways to achieve load testing in the cloud, directly and through trusted partners, allowing you a choice in addressing your specific

4. **Step 2)** In next window, enter your personal details and click on “Next” button

5. 

Software / Application Lifecycle Management / Performance Engineering / Load Testing, Software Testing Tool, LoadRunner

1 About yourself > 2 Terms of service > 3 Download

HP LoadRunner 12.00 English SW E-Media Evaluation

About yourself

First Name *

Last Name *

Email *

May HP contact you via email? * ○ Yes ● No

What are your plans for using this software? * Evaluate

About your company

Company *

Phone * 1 (Country *, Area, Phone *)

Country * United States

Address1 *

Address 2

City/town *

State/province * -- Please choose --

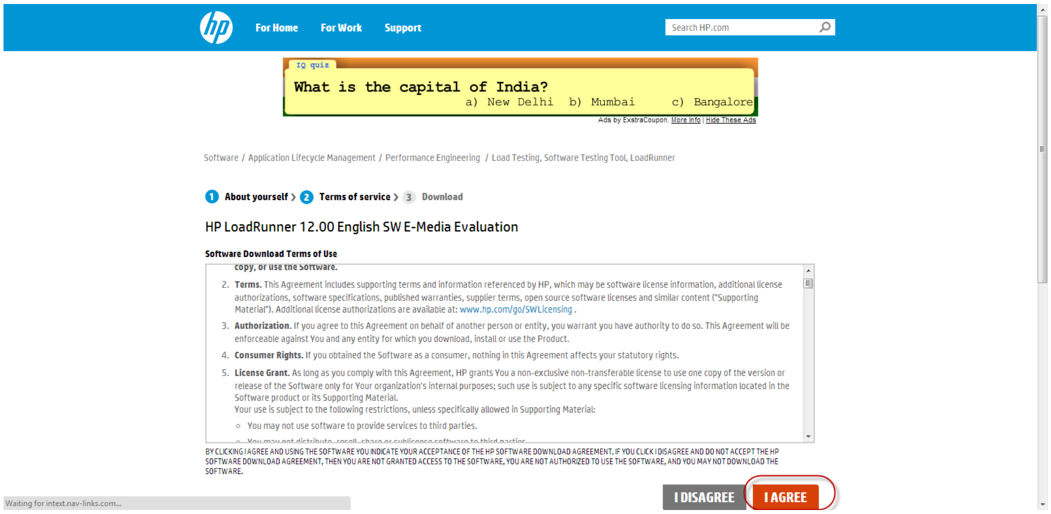
Zip/postal code *

Required *

Privacy Statement

6. **Step 3)** A window pops up showing the terms and agreement. Click on “ I Agree”

7.



Software / Application Lifecycle Management / Performance Engineering / Load Testing, Software Testing Tool, LoadRunner

1 About yourself > 2 Terms of service > 3 Download

HP LoadRunner 12.00 English SW E-Media Evaluation

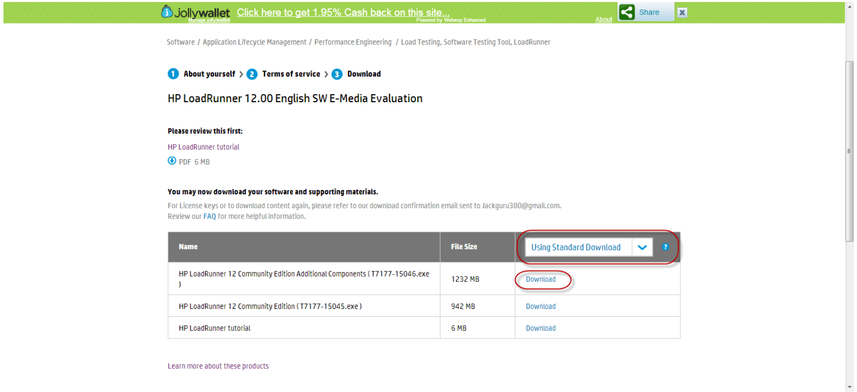
Software Download Terms of Use

copy, or use the Software.

2. **Terms.** This Agreement includes supporting terms and information referenced by HP, which may be software license information, additional license authorizations, software specifications, published warranties, supplier terms, open source software licenses and similar content ("Supporting Material"). Additional license authorizations are available at: www.hp.com/go/SWLicensing .
3. **Authorization.** If you agree to this Agreement on behalf of another person or entity, you warrant you have authority to do so. This Agreement will be enforceable against You and any entity for which you download, install or use the Product.
4. **Consumer Rights.** If you obtained the Software as a consumer, nothing in this Agreement affects your statutory rights.
5. **License Grant.** As long as you comply with this Agreement, HP grants You a non-exclusive non-transferable license to use one copy of the version or release of the Software only for Your organization's internal purposes; such use is subject to any specific software licensing information located in the Software product or its Supporting Material.
Your use is subject to the following restrictions, unless specifically allowed in Supporting Material:
   - You may not use software to provide services to third parties.
   - You may not distribute, resell, share or sublicense software to third parties.

BY CLICKING I AGREE AND USING THE SOFTWARE YOU INDICATE YOUR ACCEPTANCE OF THE HP SOFTWARE DOWNLOAD AGREEMENT. IF YOU CLICK I DISAGREE AND DO NOT ACCEPT THE HP SOFTWARE DOWNLOAD AGREEMENT, THEN YOU ARE NOT GRANTED ACCESS TO THE SOFTWARE, YOU ARE NOT AUTHORIZED TO USE THE SOFTWARE, AND YOU MAY NOT DOWNLOAD THE SOFTWARE.

8. **Step 4)** Click on "Download" on the next page as shown below.



Software / Application Lifecycle Management / Performance Engineering / Load Testing, Software Testing Tool, LoadRunner

1 About yourself > 2 Terms of service > 3 Download

HP LoadRunner 12.00 English SW E-Media Evaluation

Please review this first:

HP LoadRunner tutorial
PDF 6 MB

You may now download your software and supporting materials.

For License keys or to download content again, please refer to our download confirmation email sent to Jackguru380@gmail.com.
Review our FAQ for more helpful information.

Name	File Size	Using Standard Download
HP LoadRunner 12 Community Edition Additional Components (T7177-15046.exe)	1232 MB	Download
HP LoadRunner 12 Community Edition (T7177-15045.exe)	942 MB	Download
HP LoadRunner tutorial	6 MB	Download

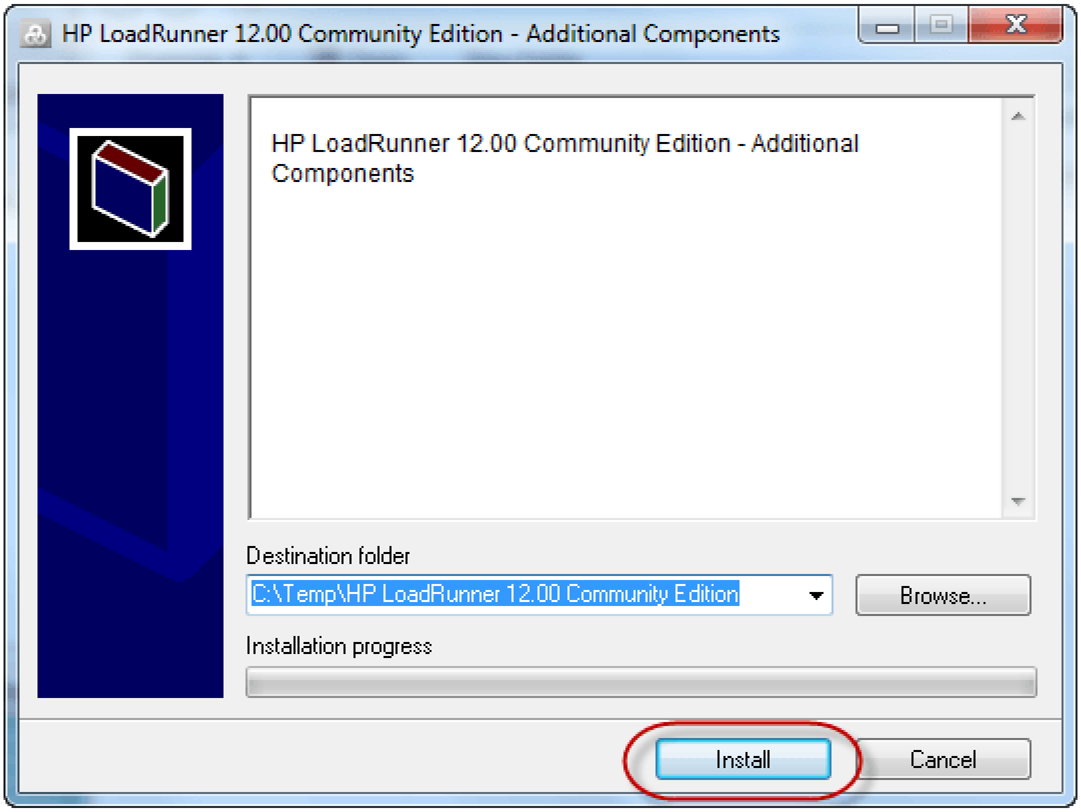
Learn more about these products

9.

10. PART 2} Installation

11. **Step 1)** Once download is complete, open the installation file. A window will open, click on install

12.



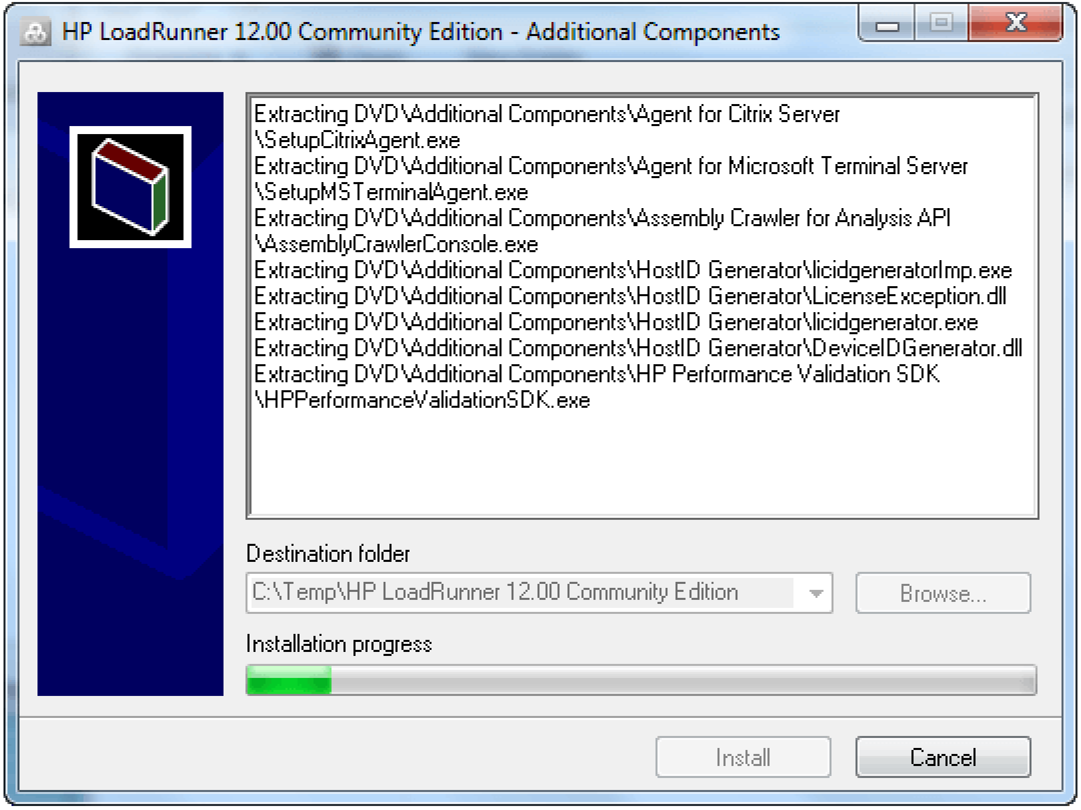
HP LoadRunner 12.00 Community Edition - Additional Components

HP LoadRunner 12.00 Community Edition - Additional Components

Destination folder: C:\Temp\HP LoadRunner 12.00 Community Edition

13.

14. **Step 2)** Once you click on install, it will start extracting the set up in "Temp" folder.



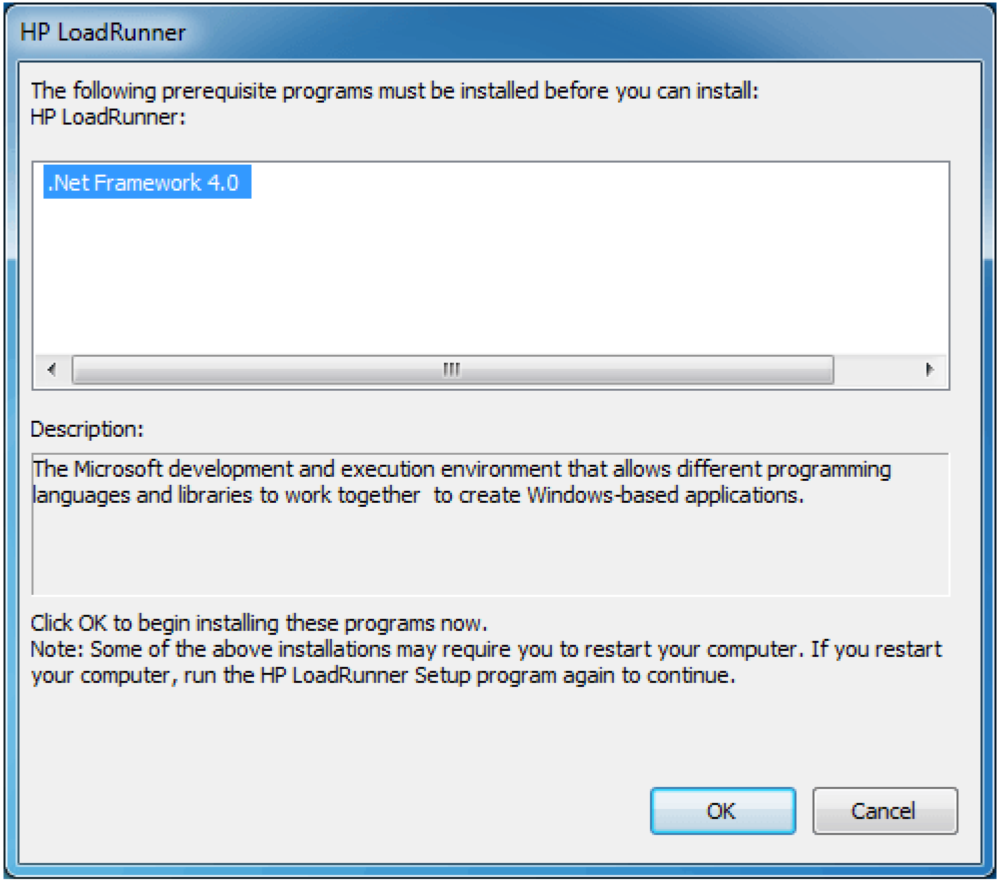
HP LoadRunner 12.00 Community Edition - Additional Components

```
Extracting DVD\Additional Components\Agent for Citrix Server
\SetupCitrixAgent.exe
Extracting DVD\Additional Components\Agent for Microsoft Terminal Server
\SetupMSTerminalAgent.exe
Extracting DVD\Additional Components\Assembly Crawler for Analysis API
\AssemblyCrawlerConsole.exe
Extracting DVD\Additional Components\HostID Generator\licidgeneratorImp.exe
Extracting DVD\Additional Components\HostID Generator\LicenseException.dll
Extracting DVD\Additional Components\HostID Generator\licidgenerator.exe
Extracting DVD\Additional Components\HostID Generator\DeviceIDGenerator.dll
Extracting DVD\Additional Components\HP Performance Validation SDK
\HPPerformanceValidationSDK.exe
```

Destination folder: C:\Temp\HP LoadRunner 12.00 Community Edition

15.

16. **Step 3)** You need to have Microsoft .net framework 4.5 installed before you begin installation. However, the LoadRunner setup will automatically prompt you to install it if you do not already have. Below is step by step guide on LoadRunner installation:

17. 

HP LoadRunner

The following prerequisite programs must be installed before you can install:
HP LoadRunner:

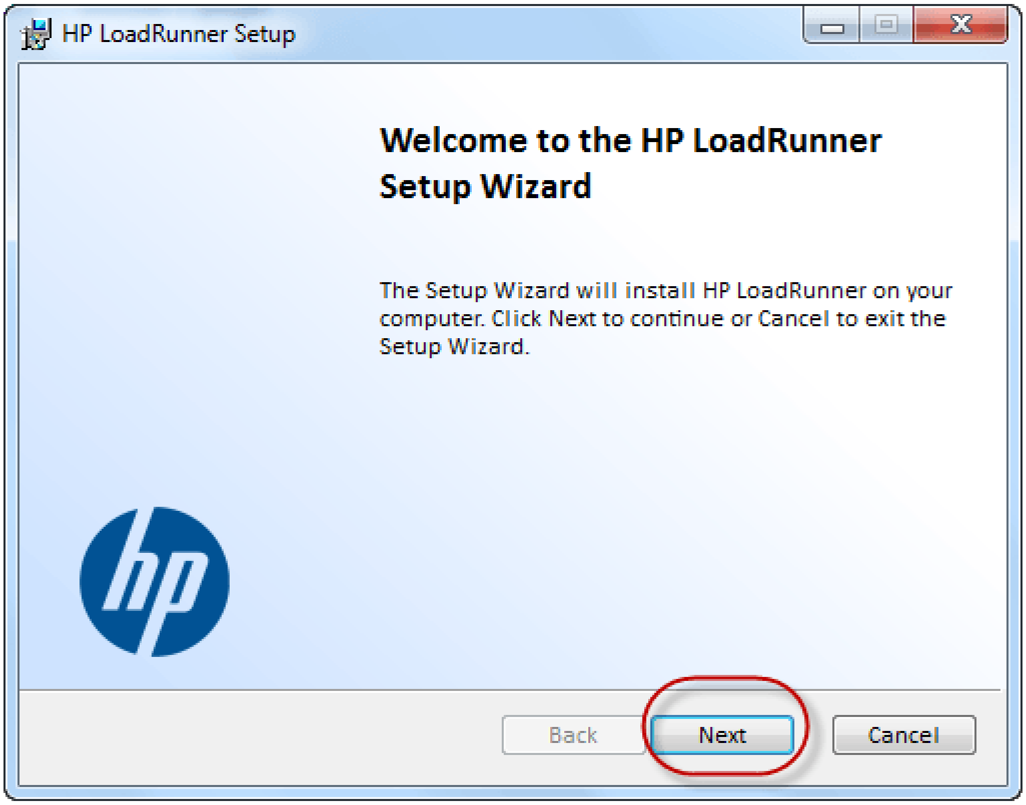
.Net Framework 4.0

Description:

The Microsoft development and execution environment that allows different programming languages and libraries to work together to create Windows-based applications.

Click OK to begin installing these programs now.
Note: Some of the above installations may require you to restart your computer. If you restart your computer, run the HP LoadRunner Setup program again to continue.

18. Click OK button to proceed with the installation of missing prerequisites. If you already have .net framework 4.0 or above, you'll not see this screen, rather the setup will show a splash screen for a moment till it calculates space requirements. Refer below for a snapshot.
19. **Step 4)** Next, HP loadrunner Setup wizard is shown. The setup will enable the Next button once it has calculated space requirement. Click Next.



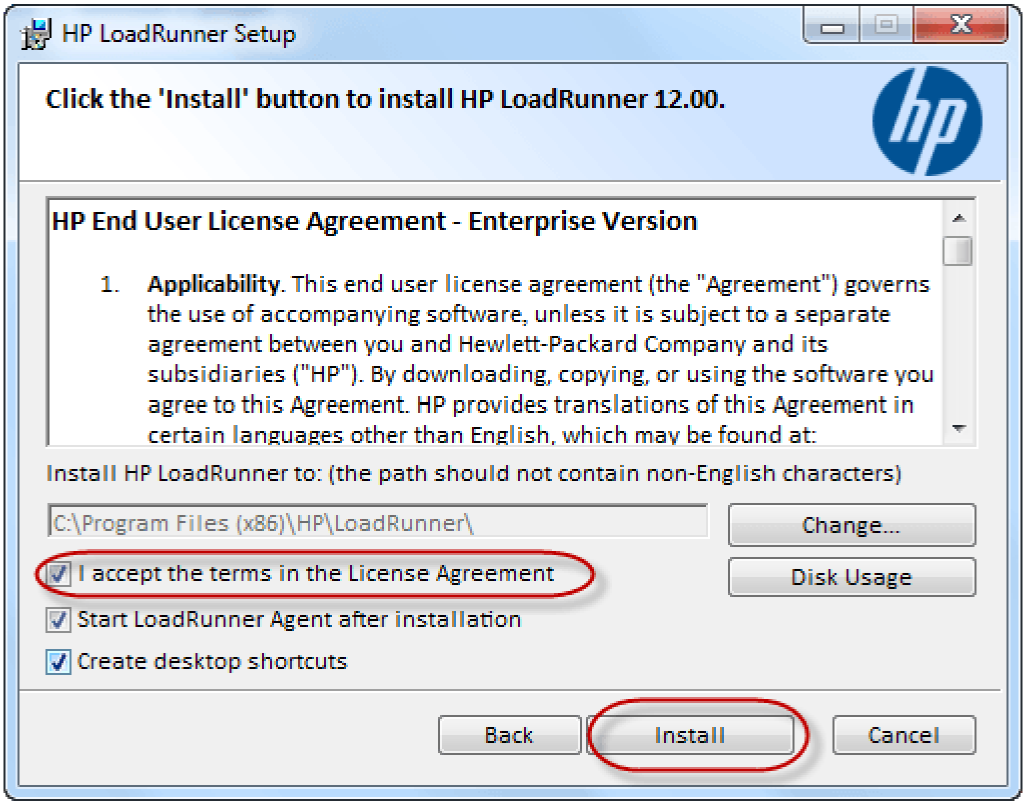
HP LoadRunner Setup

Welcome to the HP LoadRunner Setup Wizard

The Setup Wizard will install HP LoadRunner on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

20.

21. **Step 5)** In next screen, you can choose the directory you wish to install LoadRunner. Click on the check box and accept for agreement and then click on installRead the license agreement and if you agree, click Install button to begin copying of files. The setup will first validate the package to avoid disruption whilst installation progresses.



HP LoadRunner Setup

Click the 'Install' button to install HP LoadRunner 12.00.

HP End User License Agreement - Enterprise Version

1. **Applicability**. This end user license agreement (the "Agreement") governs the use of accompanying software, unless it is subject to a separate agreement between you and Hewlett-Packard Company and its subsidiaries ("HP"). By downloading, copying, or using the software you agree to this Agreement. HP provides translations of this Agreement in certain languages other than English, which may be found at:

Install HP LoadRunner to: (the path should not contain non-English characters)

C:\Program Files (x86)\HP\LoadRunner\

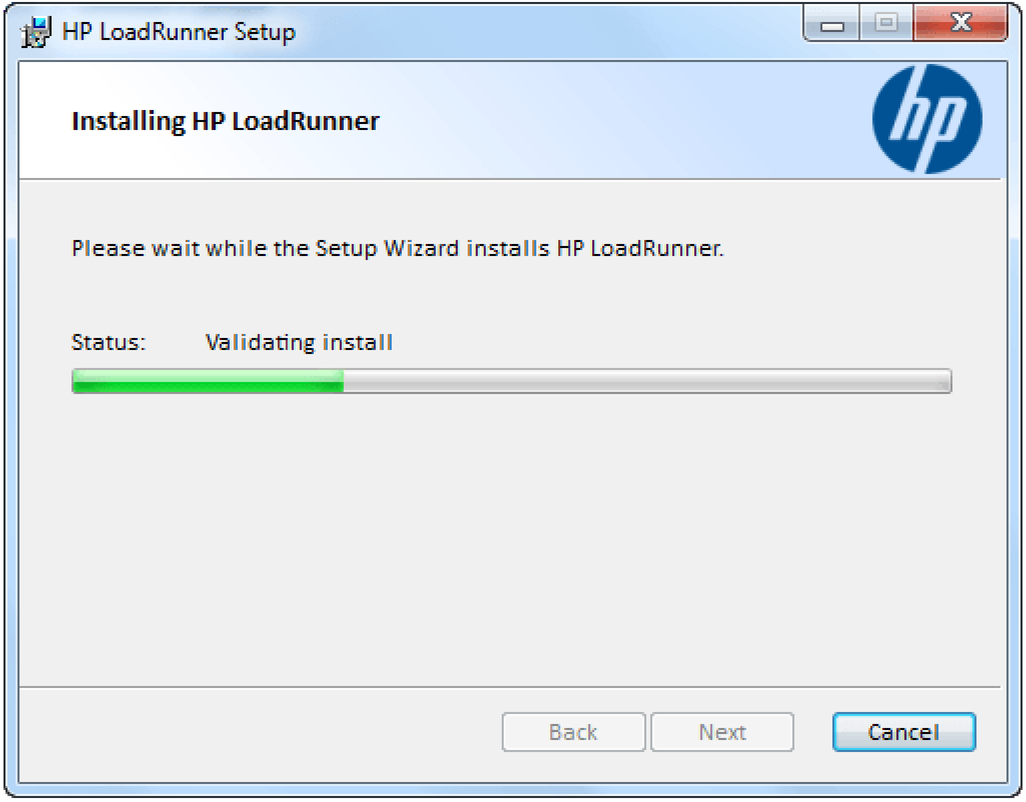
☑ I accept the terms in the License Agreement

☑ Start LoadRunner Agent after installation

☑ Create desktop shortcuts

22. `

23. **Step 6)** It will take few seconds to install LoadRunner After validating the package, the setup will copy all files in directory selected earlier. It will register all components with the system and acknowledge upon completion. The setup will then show Finish button.



HP LoadRunner Setup

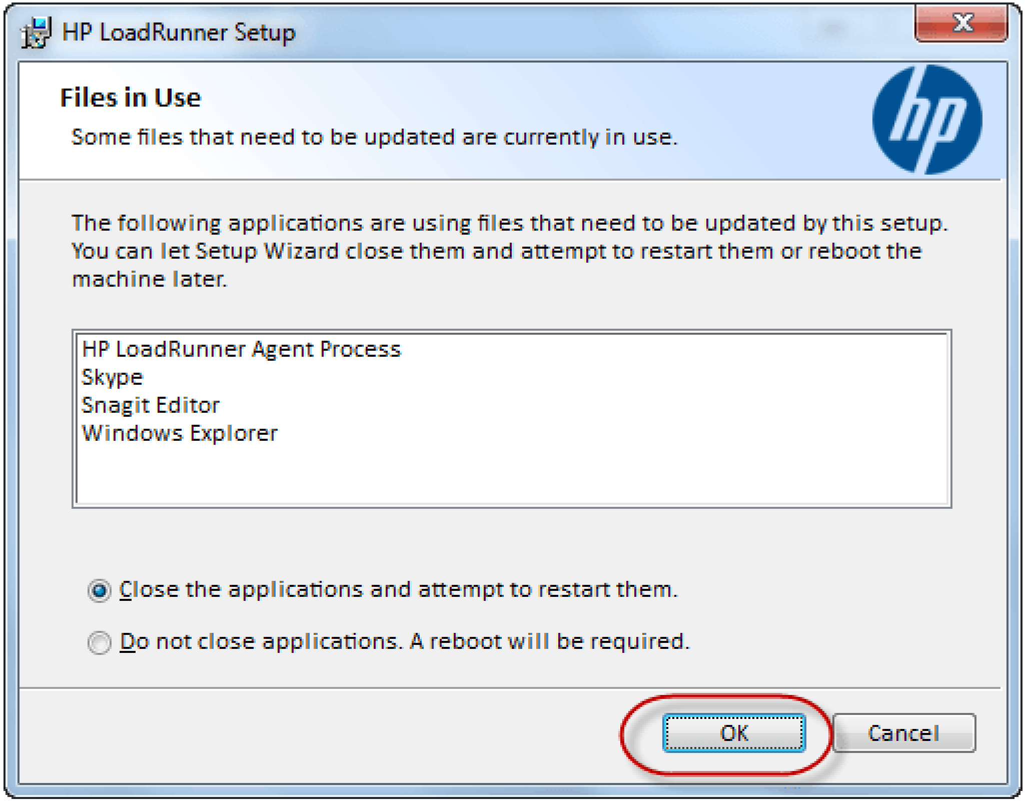
Installing HP LoadRunner

Please wait while the Setup Wizard installs HP LoadRunner.

Status: Validating install

24.

25. **Step 7)** In next step, click on “OK” to proceed.



HP LoadRunner Setup

Files in Use

Some files that need to be updated are currently in use.

The following applications are using files that need to be updated by this setup. You can let Setup Wizard close them and attempt to restart them or reboot the machine later.

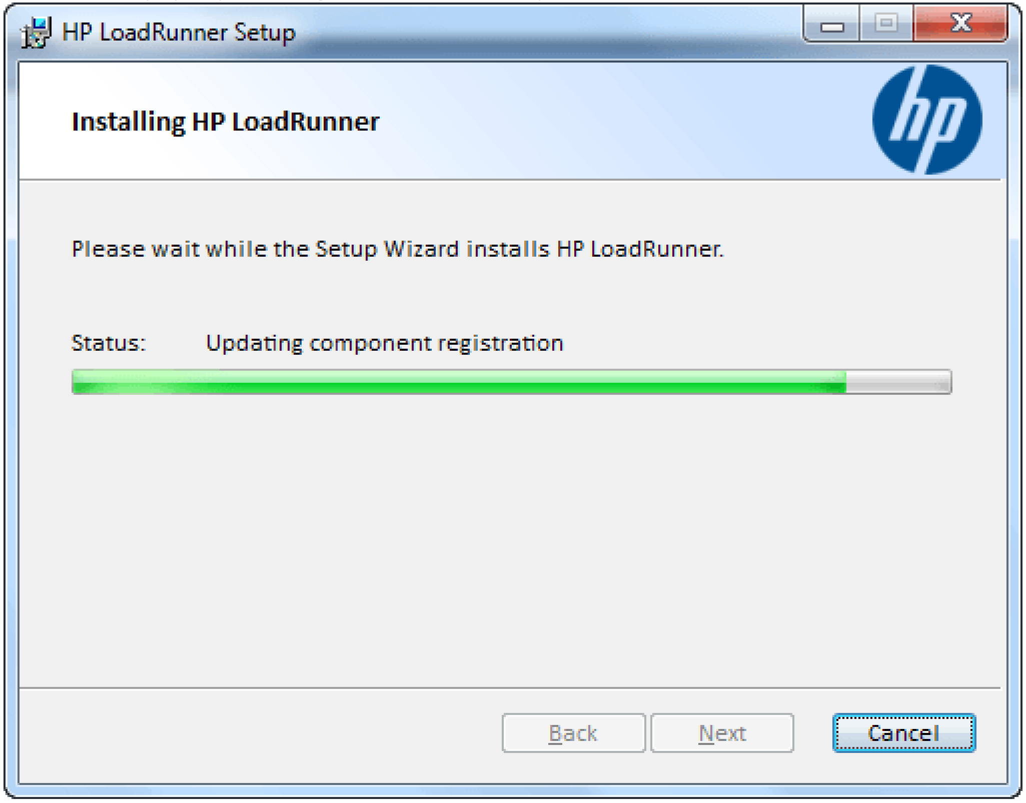
HP LoadRunner Agent Process
Skype
Snagit Editor
Windows Explorer

● Close the applications and attempt to restart them.

○ Do not close applications. A reboot will be required.

26.

27. **Step 8)** A window pops up showing installation of Setup Wizard



HP LoadRunner Setup

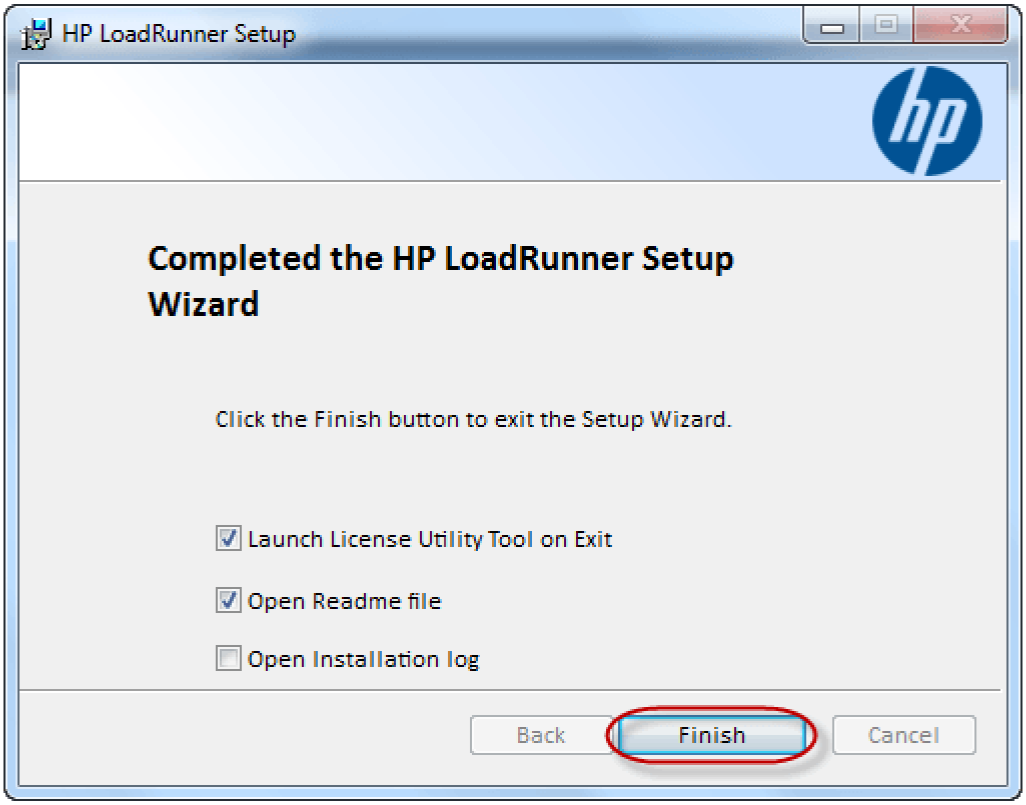
Installing HP LoadRunner

Please wait while the Setup Wizard installs HP LoadRunner.

Status: Updating component registration

28.

29. **Step 9)** Once the LoadRunner setup wizard is installed, click on open installation log and then click on finish as shown below



HP LoadRunner Setup

Completed the HP LoadRunner Setup Wizard

Click the Finish button to exit the Setup Wizard.

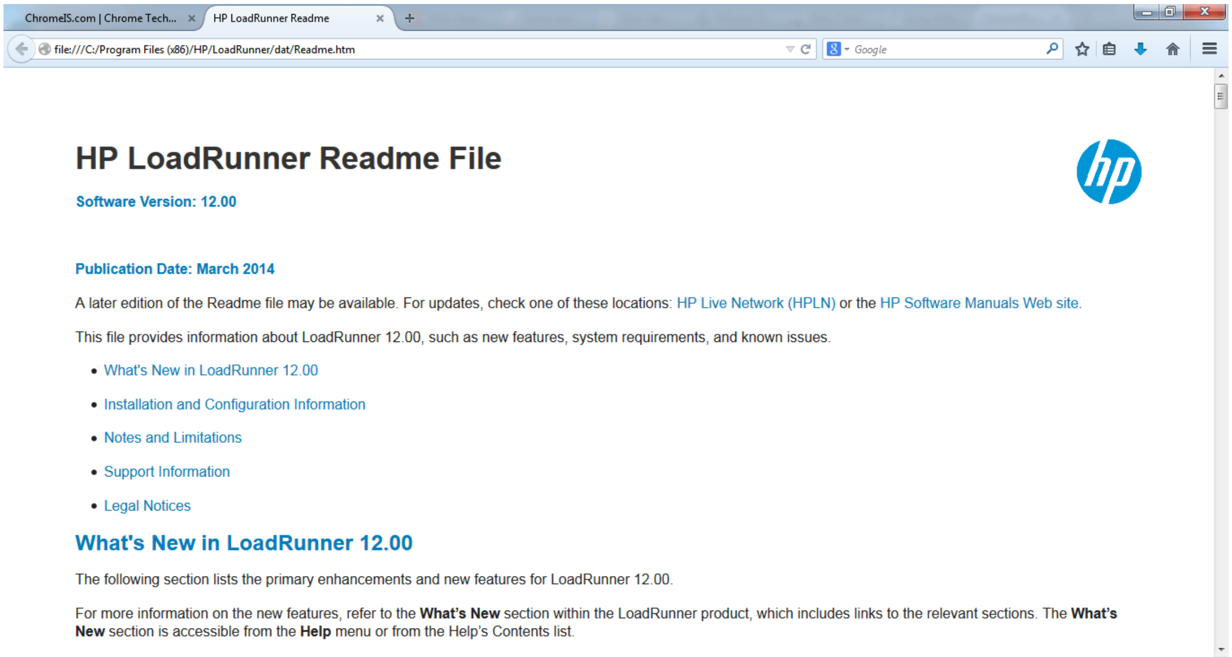
☑ Launch License Utility Tool on Exit

☑ Open Readme file

□ Open Installation log

30.

31. **Step 10)** Once finished, the setup will open readme file by default which contains useful information. Whether you're new to LoadRunner of experience user, it is best recommended to go through readme file at least once.



ChromeIS.com | Chrome Tech...
HP LoadRunner Readme
file:///C:/Program Files (x86)/HP/LoadRunner/dat/Readme.htm
Google

HP LoadRunner Readme File

Software Version: 12.00

Publication Date: March 2014

A later edition of the Readme file may be available. For updates, check one of these locations: HP Live Network (HPLN) or the HP Software Manuals Web site.

This file provides information about LoadRunner 12.00, such as new features, system requirements, and known issues.

- What's New in LoadRunner 12.00
- Installation and Configuration Information
- Notes and Limitations
- Support Information
- Legal Notices

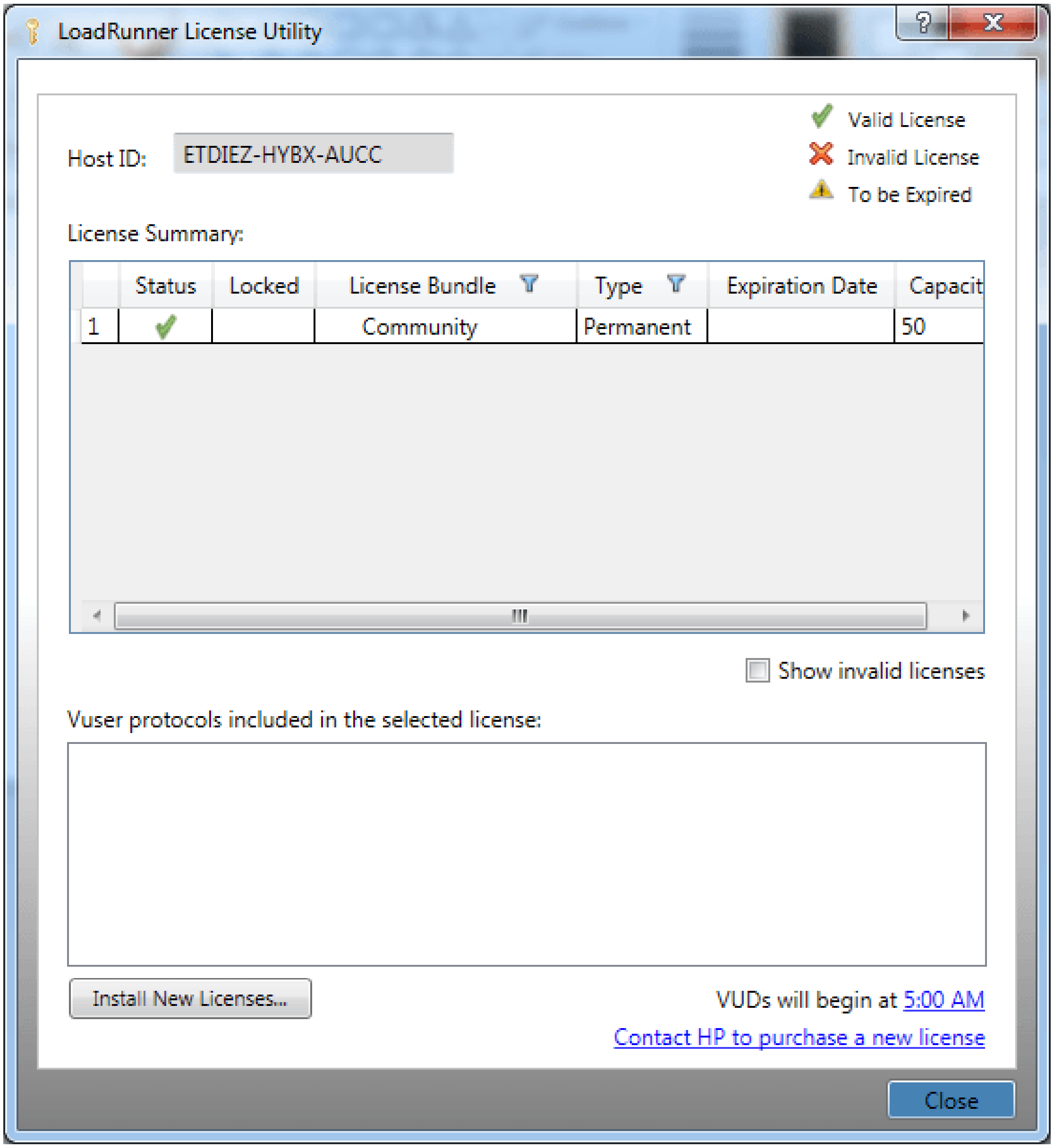
What's New in LoadRunner 12.00

The following section lists the primary enhancements and new features for LoadRunner 12.00.

For more information on the new features, refer to the **What's New** section within the LoadRunner product, which includes links to the relevant sections. The **What's New** section is accessible from the **Help** menu or from the Help's Contents list.

32.

33. **Step 11)** The LoadRunner installation is completed at this point. However, you can see the setup has opened License Utility. Observe this screen and ensure there is at least 1 license installed and its status has a tick mark in green as shown in below screenshot. Starting with LoadRunner 12, HP has issued free license of 50 VUsers which has no expiry.



Host ID: ETDIEZ-HYBX-AUCC

✓ Valid License
✗ Invalid License
⚠ To be Expired

License Summary:

	Status	Locked	License Bundle	Type	Expiration Date	Capacit
1	✓		Community	Permanent		50

□ Show invalid licenses

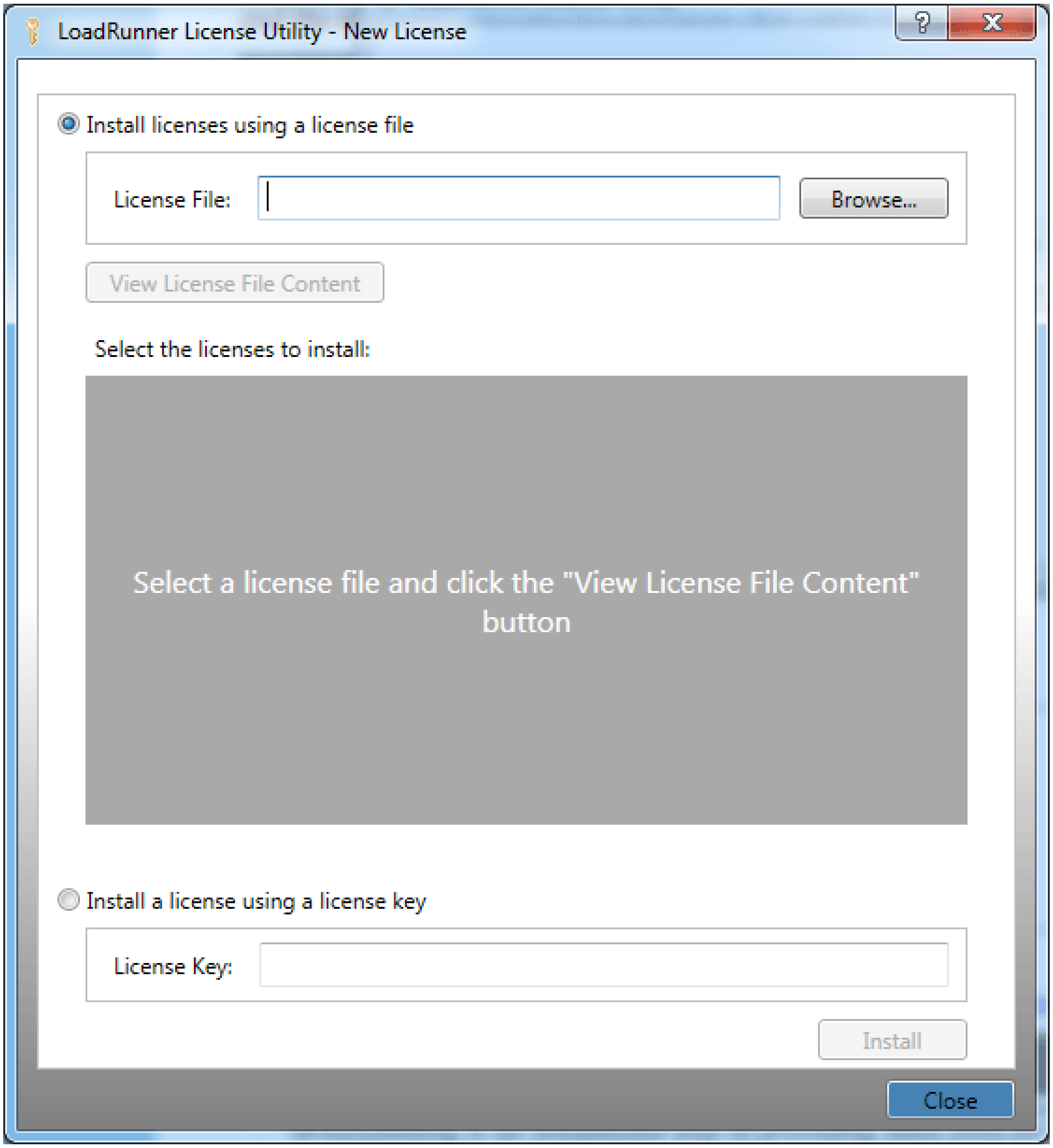
Vuser protocols included in the selected license:

VUDs will begin at 5:00 AM

Contact HP to purchase a new license

34.

35. If you've a valid license from HP, you can install it by clicking on Install New License button. The new window will open where license file can be browsed and imported – or otherwise a license key can be directly pasted in the bottom text box. Refer to below snapshot:

36. 

LoadRunner License Utility - New License
● Install licenses using a license file
License File:
Browse...
View License File Content
Select the licenses to install:
○ Install a license using a license key
License Key:
Install
Close

Select a license file and click the "View License File Content" button

37. Installation of the new version is optimized and takes less than 10 minutes on a Core i5 machine with 8GB RAM. However the minimum hardware requirements are different, as put below:

Component	**Minimum Requirement**
Processor speed	1.6 GHZ or faster
Operating system	Works on: Windows Server 2008 R2 SP1 64-bit Windows Server 2012 64-bit [only supported for Controller and Analysis] Windows 7 SP1 32 (supported) or 64-bit (recommended) Windows 8 64-bit
Memory (RAM)	**Minimum:**2 GB **Recommended:**4 GB or higher
Screen resolution	**Minimum:**1024 x 768
Browser	Microsoft Internet Explorer 8, 9 and 10
Available hard disk space	**Minimum:**40 GB

38. **Minimum Hardware requirements for LoadRunner**

39. A Controller simulates load through Load Generators and these are configured on separate machines. HP had recommended minimum hardware requirements for Load Injectors:

Component	**Minimum Requirement**
Processor speed	1.6 GHZ or faster
Operating system	Works on: Windows Server 2008 R2 SP1 64-bit Windows Server 2012 64-bit Windows 7 SP1 32 (supported) or 64-bit (recommended) Windows 8 64-bit
Memory (RAM)	**Minimum:**1 GB for Load Generator or 2GB for MI Listener **Recommended:**4 GB or higher
Screen resolution	**Minimum:**1024 x 768
Browser	Microsoft Internet Explorer 8, 9, 10 or 11
Available hard disk space	**Minimum:**40 GB

Introduction to HP LoadRunner and its ArchitectureLoadRunner is a Performance Testing tool which was pioneered by Mercury in 1999. Loadrunner was later acquired by HP in 2009.
LoadRunner **supports various development tools, technologies and communication protocols**. In fact this is the only tool in market which **supports** such **large number of protocols** to conduct performance testing.
LoadRunner is not only pioneer tool in Performance Testing, but it is still market leader in the Performance testing paradigm. In a recent assessment, LoadRunner has about **85% market share** in Performance Testing industry.

Broadly, LoadRunner supports RIA (Rich Internet Applications), Web 2.0 (HTTP/HTML, Ajax, Flex and Silverlight etc.), Mobile, SAP, Oracle, MS SQL Server, Citrix, RTE, Mail and above all, Windows Socket. There is no competitor tool in the market which could offer such wide variety of protocols vested in single tool.

What is more convincing to pick LoadRunner for performance testing is the credibility of this tool. LoadRunner has long established reputation as often you will find **clients cross verifying your performance benchmarks using LoadRunner.** You'll find relief if you're already using LoadRunner for your performance testing needs.
LoadRunner is tightly integrated with other HP Tools like Unified Functional Test (QTP) & ALM (Application Lifecycle Management) with empowers you to perform your end to end Testing Processes.

LoadRunner works on a principal of simulating Virtual Users on the subject application. These Virtual Users, also termed as VUsers, replicate client's requests and expect corresponding response to pass a transaction.

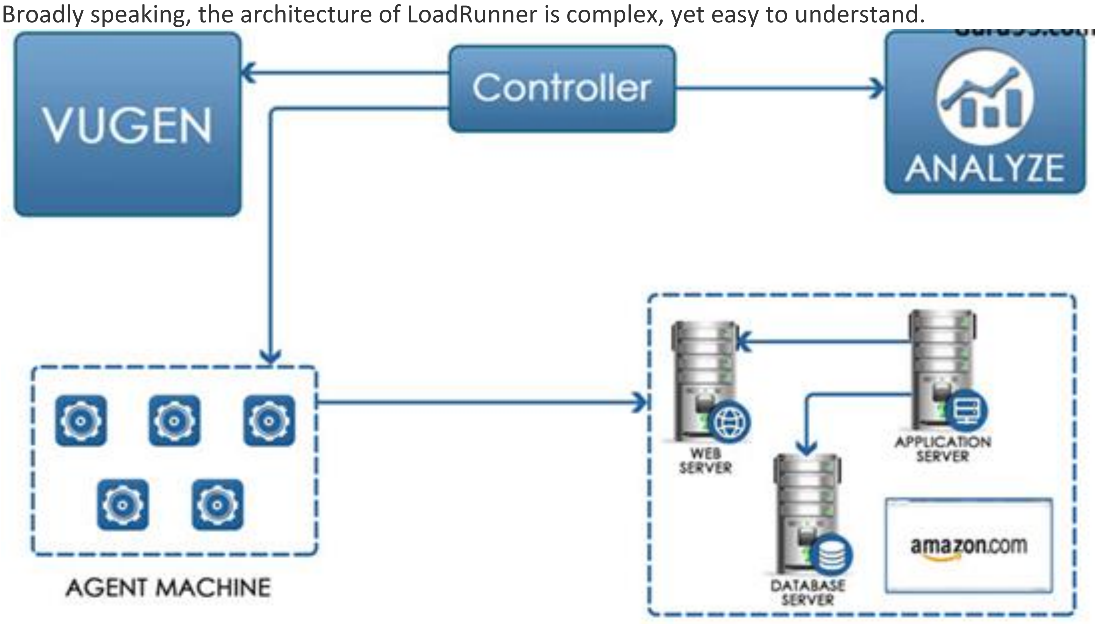
Why do you need Performance Testing?

An estimated **loss of 4.4 billion in revenue** is recorded annually due to poor web performance.
In today's age of Web 2.0, users click away if a website doesn't respond within 8 seconds. Imagine yourself waiting for 5 seconds when searching over Google or making a friend request on Facebook. The repercussions of performance downtime are often more devastating than ever imagined. We've examples such as those that recently hit Bank of America Online Banking, Amazon Web Services, Intuit or Blackberry.
According to Dunn & Bradstreet, 59% of Fortune 500 companies experience an estimated 1.6 hours of downtime every week. Considering the average Fortune 500 company with a minimum of 10,000 employees is paying $56 per hour, the labour part of downtime costs for such an organization would be $896,000 weekly, translating into more than $46 million per year.
Only a 5 minute downtime of Google.com (19-Aug-13) is estimated to cost the search giant as much as $545,000.
Its estimate that companies lost sales worth $1100 per second due to a recent Amazon Web Service Outage.
When a software system is deployed by an organization, it may encounter many scenarios that possibly result in performance latency. A number of factors cause decelerating performance, few example may include:

- Increased number of records present in the database
- Increased number of simultaneous requests made to the system
- larger number of users accessing the system at a time as compared to the past

What is LoadRunner Architecture?

Broadly speaking, the architecture of LoadRunner is complex, yet easy to understand.



Suppose you are assigned to check performance of Amazon.com for 5000 users
In a real life situation, these all these 5000 users will not be at homepage but in different section of the websites. How can we simulate different

VUGen:

VUGen or Virtual User Generator is an IDE (Integrated Development Environment) or a rich coding editor. VUGen is used to replicate System Under Load (SUL) behaviour. VUGen provides a "recording" feature which records communication to and from client and Server in form of a coded script - also called VUser script.
So considering the above example, VUGen can record to simulate following business processes:

1. Surfing the Products Page of Amazon.com
2. Checkout
3. Payment Processing
4. Checking MyAccount Page

Controller

Once a VUser script is finalized, Controller is the main component which controls the Load simulation by managing, for example:

- How many VUsers to simulate against each business process or VUser Group
- Behaviour of VUsers (ramp up, ramp down, simultaneous or concurrent nature etc.)
- Nature of Load scenario e.g. Real Life or Goal Oriented or verifying SLA
- Which injectors to use, how many VUsers against each injector
- Collate results periodically
- IP Spoofing
- Error reporting
- Transaction reporting etc.

Taking analogy from our example controller will add following parameter to the VUGen Script

1) **3500 Users are** Surfing the Products Page of Amazon.com
2) **750 Users are in** Checkout
3) **500 Users are** performing Payment Processing
4) **250 Users are** Checking MyAccount Page ONLY after 500 users have done Payment Processing

Even more complex scenarios are possible

1. Initiate 5 VUsers every 2 seconds till a load of 3500 VUsers (surfing Amzon product page) is achieved.
2. Iterate for 30 minutes
3. Suspend iteration for 25 VUsers
4. Re-start 20 VUSers
5. Initiate 2 users (in Checkout, Payment Processing , MyAccounts Page) every second.
6. 2500 VUsers will be generated at Machine A
7. 2500 VUsers will be generated at Machine B

Agents Machine/Load Generators/Injectors

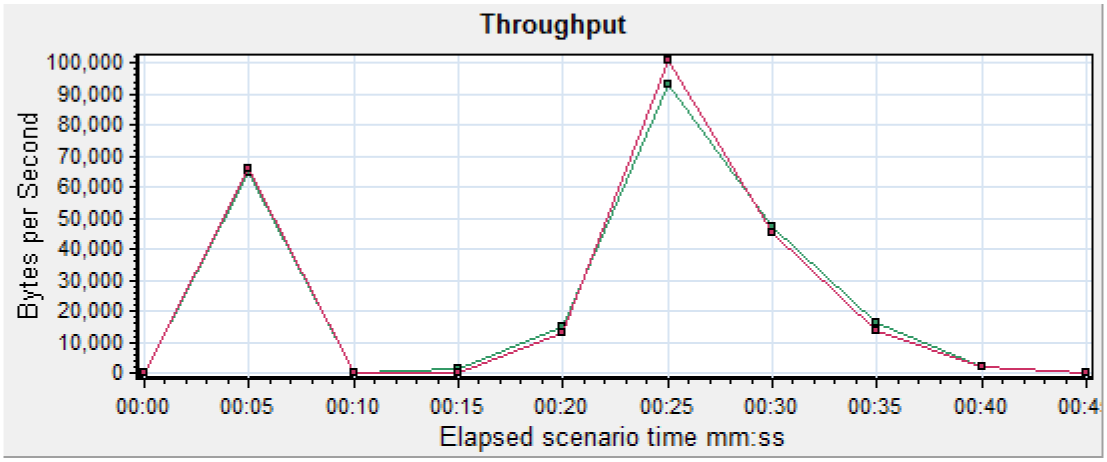
LoadRunner Controller is responsible to simulate thousands of VUsers - these VUsers consume hardware resources for example processor and memory - hence putting a limit on the machine which is simulating them. Besides, Controller simulates these VUsers from the same machine (where Controller resides) & hence the results may not be precise. To address this concern, all VUsers are spread across various machines, called **Load Generators or Load Injectors.**

As a general practice, Controller resides on a different machine and load is simulated from other machines. Depending upon the protocol of VUser scripts and machine specifications, a number of Load Injectors may be

required for full simulation. For example, VUsers for an HTTP script will require 2-4MB per VUser for simulation, hence 4 machines with 4 GB RAM each will be required to simulate a load of 10,000 VUsers.
Taking Analogy from our Amazon Example, the output of this component will be

Analysis:

Once Load scenarios have been executed, the role of "**Analysis**" component comes in.
During the execution, Controller creates a dump of results in raw form & contains information like, which version of LoadRunner created this results dump and what were configurations.
All the errors and exceptions are logged in a Microsoft access database, named, output.mdb. The "Analysis" component reads this database file to perform various types of analysis and generates graphs.
These graphs show various trends to understand the reasoning behind errors and failure under load; thus help figuring whether optimization is required in SUL, Server (e.g. JBoss, Oracle) or infrastructure.
Below is an example where bandwidth could be creating bottleneck. Let's say Web server has 1GBps capacity whereas the data traffic exceeds this capacity causing subsequent users to suffer. To determine system caters to such needs, Performance Engineer needs to analyse application behaviour with abnormal load. Below is a graph LoadRunner generates to elicit bandwidth.



Performance Testing Roadmap: Detailed Steps
Performance Testing Roadmap can be broadly divided into 5 steps:

- Planning for Load Test
- Create VUGen Scripts
- Scenario Creation
- Scenario Execution
- Results Analysis (followed by system tweaking)

Now that you've LoadRunner installed, let's understand the steps involved in the process one by one.

Planning for the Load Test
Planning for Performance Testing is different from planning an SIT (System Integration Testing) or UAT (User Acceptance Testing). Planning can be further divided into small stages as described below:
Assemble Your Team

When getting started with Performance Testing, it is best to document who will be participate in the activity from each team involved during the process.

Project Manager:
Nominate the project manager who will own this activity and serve as point person for escalation.

Function Expert/ Business Analyst:
Provide Usage Analysis of SUL & provides expertise on business functionality of website/SUL

Performance Testing Expert:

Creates the automated performance tests and executes load scenarios
System Architect:
Provides blueprint of the SUL

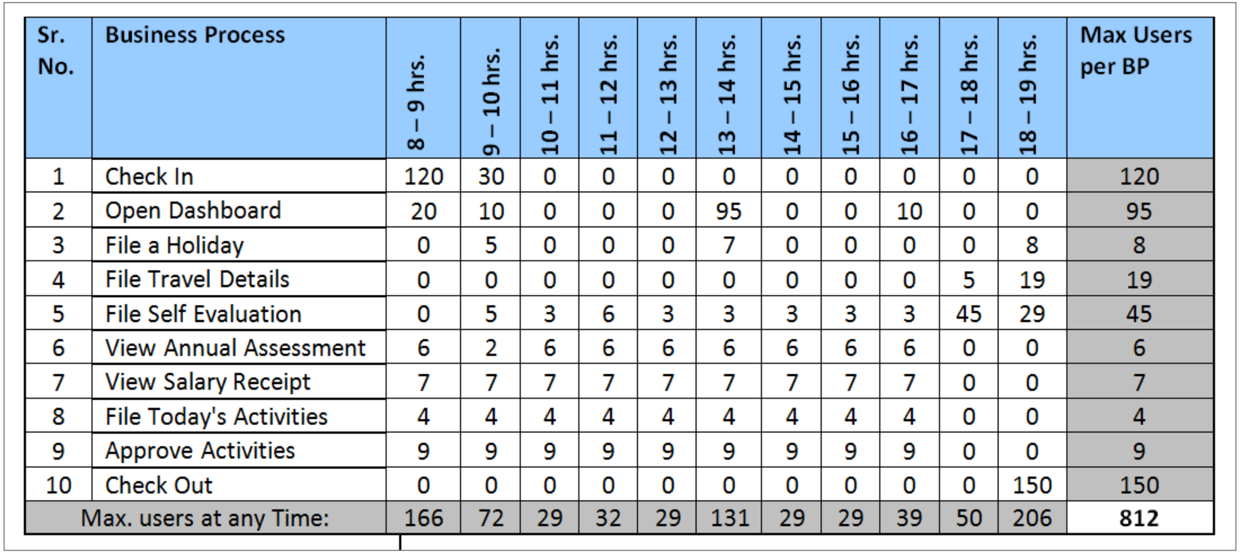
Web Developer and SME:

- Maintains website & provide monitoring aspects
- Develops website and fixes bugs

System Administrator:

- Maintains involved servers throughout testing project

Outline applications and Business Processes involved:
Successful load testing requires that you plan to carry out certain business process. A Business Process consists of clearly defined steps in compliance to desired business transactions - so as to accomplish your load testing objectives.
A requirements metric can be prepared to elicit user load on the system. Below is an example from an attendance system in a company:



Sr. No.	Business Process	8 – 9 hrs.	9 – 10 hrs.	10 – 11 hrs.	11 – 12 hrs.	12 – 13 hrs.	13 – 14 hrs.	14 – 15 hrs.	15 – 16 hrs.	16 – 17 hrs.	17 – 18 hrs.	18 – 19 hrs.	Max Users per BP
1	Check In	120	30	0	0	0	0	0	0	0	0	0	120
2	Open Dashboard	20	10	0	0	0	95	0	0	10	0	0	95
3	File a Holiday	0	5	0	0	0	7	0	0	0	0	8	8
4	File Travel Details	0	0	0	0	0	0	0	0	0	5	19	19
5	File Self Evaluation	0	5	3	6	3	3	3	3	3	45	29	45
6	View Annual Assessment	6	2	6	6	6	6	6	6	6	0	0	6
7	View Salary Receipt	7	7	7	7	7	7	7	7	7	0	0	7
8	File Today's Activities	4	4	4	4	4	4	4	4	4	0	0	4
9	Approve Activities	9	9	9	9	9	9	9	9	9	0	0	9
10	Check Out	0	0	0	0	0	0	0	0	0	0	150	150
Max. users at any Time:		166	72	29	32	29	131	29	29	39	50	206	**812**

In the above example, the figures mention the number of users connected to the application (SUL) at given hour. We can extract the maximum number of users connected to a business process at any hour of the day which is calculated in the right most columns.
Similarly, we can conclude the total number of users connected to the application (SUL) at any hour of the day. This is calculated in the last row.
The above 2 facts combined give us the total number of users with which we need to test the system for performance.
Define Test Data Management Procedures
Statistics and observations drawn from Performance Testing are greatly influenced by numerous factors as briefed earlier. It is of critical significance to prepare Test Data for Performance Testing. Sometimes, a particular business process consumes a data set and produces a different data set. Take below example:

- A user 'A' creates a financial contract and submits it for review.
- Another user 'B' approves 200 contracts a day created by user 'A'
- Another user 'C' pays about 150 contracts a day approved by user 'B'

In this situation, User B need to have 200 contracts 'created' in the system. Besides, user C needs 150 contracts as "approved" in order to simulate load of 150 users.

This implicitly means, that you must create at least 200+150= 350 contracts.

After that, approve 150 contracts to serve as Test data for User C - the remaining 200 contracts will serve as Test Data for User B.

Outline Monitors

Speculate each and every factor which could possible affect the performance of system. For example, having reduced hardware will have potential impact on the SUL(System Under Load) performance.

Enlist all factors and setup monitors so you can gauge them. Here are few examples:

- Processor   (for Web Server, Application Server, Database Server and Injectors)
- RAM   (for Web Server, Application Server, Database Server and Injectors)
- Web/App Server (for example IIS, JBoss, Jaguar Server, Tomcat etc)
- DB Server (PGA and SGA size in case of Oracle and MSSQL Server, SPs etc.)
- Network bandwidth utilization
- Internal and External NIC in case of clustering
- Load Balancer (and that it is distributing load evenly on all nodes of clusters)
- Data flux   (calculate how much data moves to and from client and server - then calculate if capacity of NIC is sufficient to simulate X number of users)

Create VUGen Scripts

Next step after planning is to create VUser scripts.

Scenario Creation

Next step is to create your Load Scenario

Scenario Execution

Scenario execution is where you emulate user load on the server 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.

This execution may result the server to go under stress and behave abnormally. This is the very purpose of the performance Testing. The results drawn are then used for detailed analysis and root cause identification.

Results Analysis (followed by system tweaking)

During scenario execution, LoadRunner records the performance of the application under different loads. The statistics drawn from test execution are saved and details analysis is performed. The 'HP Analysis' tool generates various graphs which help in identifying the root causes behind lag of system performance, as well as system failure.

Some of the graphs obtained include:

- Time to First buffer
- Transaction Response Time
- Average Transaction Response Time
- Hits Per Second
- Windows Resources
- Errors Statistics
- Transaction Summary

Understanding VUGen in LoadRunner

VUGen is one of the four core components of LoadRunner. It is the first component you interact with when getting started with Performance Testing using HP LoadRunner.

Let’s understand a few salient features related to VUGen

VUScripts: As already described, the purpose of VUGen is to create VUScripts that are used to simulate a real-like virtual-user.

Actions: An action is set of user transactions performed in the System Under Load to achieve a defined task. An action can be compared to a function in other programming languages. Each Vuser script has 3 default functions
Vuser_init(used to login into the application)
Action (used to record the business process)
Vuser_end (used to logout of the application)
VUGen not only records scripts but also replays them (for only 1 VUser) to ensure script is recorded correctly. Once you ensure the script is recorded correctly you incorporate it in a LoadRunner scenario
In this tutorial we are going to study
1 The script development process in VUGen
2 Compare recording between QTP and LoadRunner

- Introduction to Web Tours Application
- Understanding the VUGen Environment

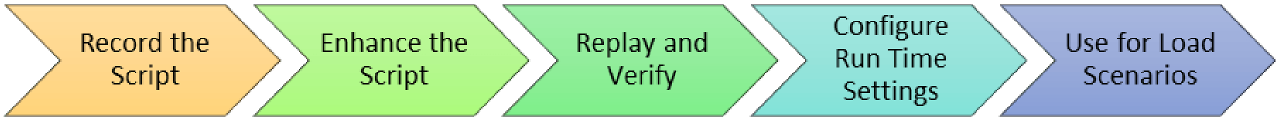
3 Creating Your First VUGen Script

- Using Transactions at Record Time
- Insert Comments at Record Time
- Code Generation
- Deciding a Protocol and Protocol Advisor
- Recording Options

4 Playback a Script and understanding Log

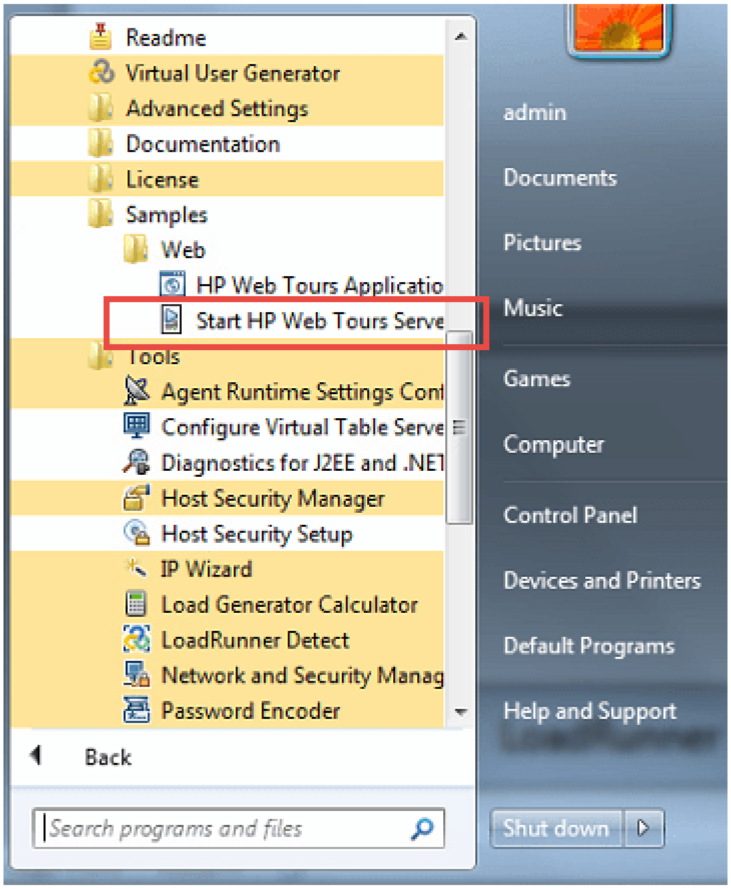
5 Overview of Files Generated During Record & Playback

The script development process in VUGen

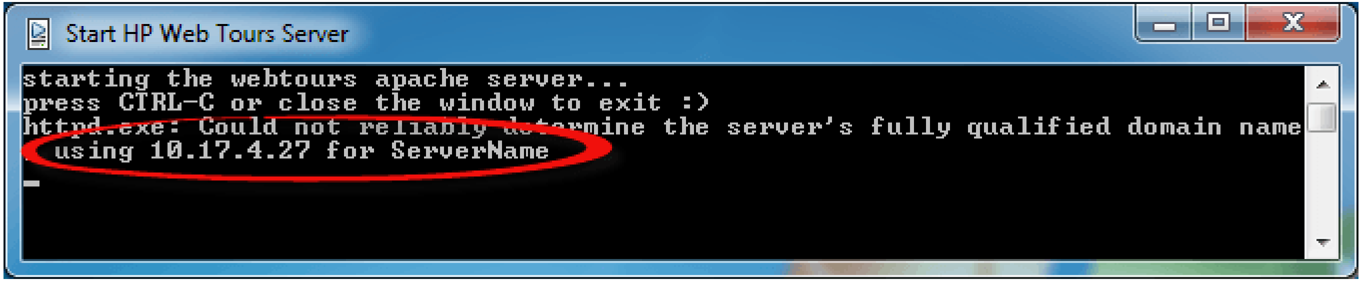


1. Record the Script: Usually this is the first step of scripting where every user action is recorded into a script.
2. Replay and Verify: Once the script is recorded, reply the script to ensure its working right. Verify any impact through application frontend or database.
3. Enhance the Script: Once recording has been verified, enhance script by adding checkpoints, validating data, adding transactions and rendezvous points.
4. Replay and Verify: As earlier, re-play your script and verify that everything is working as intended.
5. Configure Runtime Settings: Configure and control pacing duration, think time variation, proxy settings and whether you wish to ignore any external resources.
6. Use for Load Scenarios: Formulate load scenarios based on test objectives. Use load distribution and geo-wide agents to make real like scenarios.
Compare recording between QTP and LoadRunner
VUGen disregards UI Objects:
Unlike QTP scripts, LoadRunner scripts are independent of GUI. This is because the code generated does not work based on UI objects, rather, it works on the principal of client's request to the server and expecting server response back to the client. This is why replaying LoadRunner scripts are always browser independent. The VUser can communicate directly with a server by executing calls to the server API-without relying on client software (SUL) this means there will no impact on scripts if the changes are made to the UI of the System Under Load. This tells; the performance testing is completely based on client/server communication (request and response) and not the GUI objects.
Please refer to below link for more details: http://youtu.be/FDORUeqpNJs?t=3m41s
We will see in more details how VUGen works below.
Introduction to Web Tours Application
For all hands-on purpose, we will be using the Web Tours Application that comes bundled with LoadRunner.

The HP Web Tours requires a dependent "Web tours apache server" to be running before it can operate. To run the Web Tours Apache Server, go to Start Menu => HP Software => Samples => Web and then click on Start HP Web Tour Server.

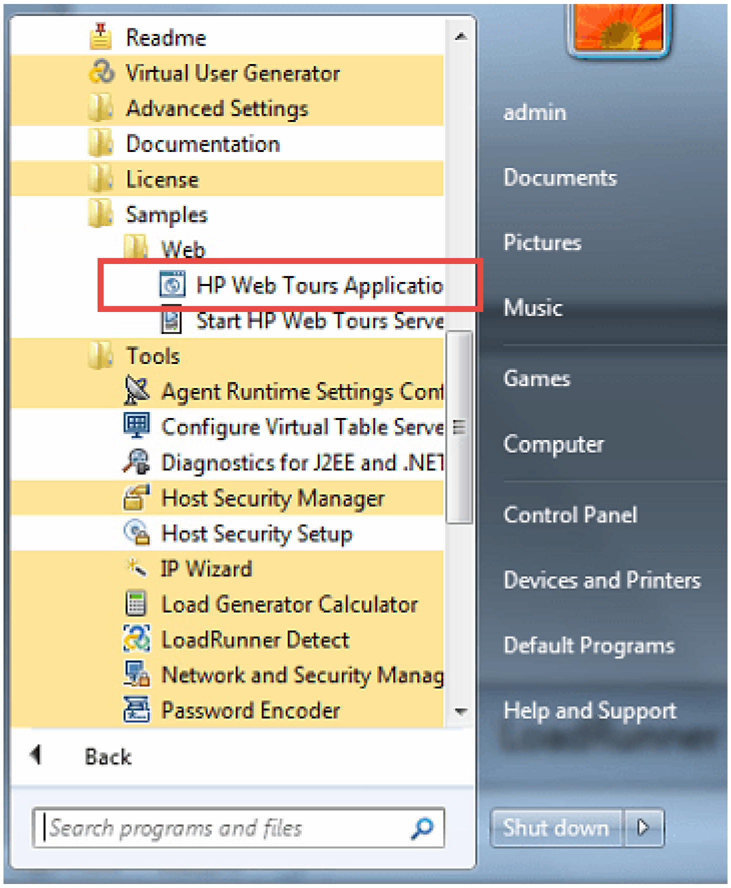


Once the "server" has been activated, it will open a shell and output on the console.
The console should look like snapshot below, look for your IP address in the shell:

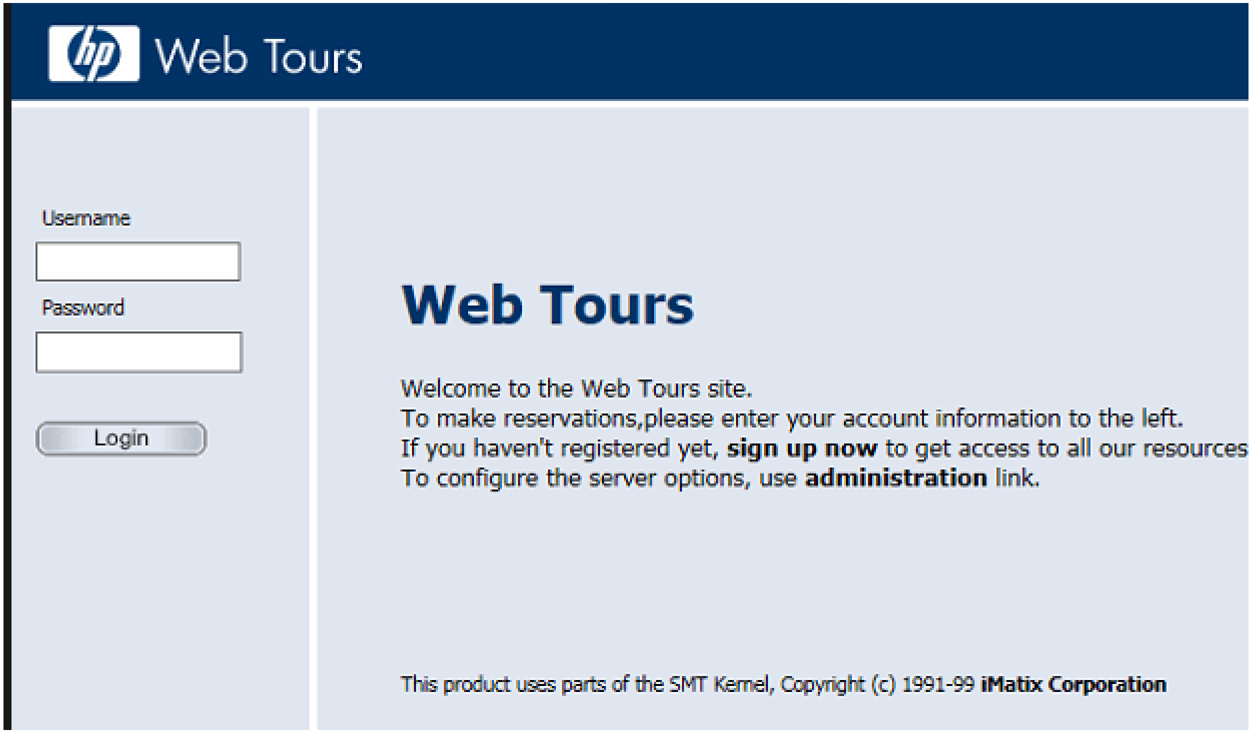


```
starting the webtours apache server...
press CTRL-C or close the window to exit :)
httpd.exe: Could not reliably determine the server's fully qualified domain name
 using 10.17.4.27 for ServerName
_
```

NOTE: if you close this window, the server will stop.
Now you can access the Web Tours Application –



Once launched, it will look like –



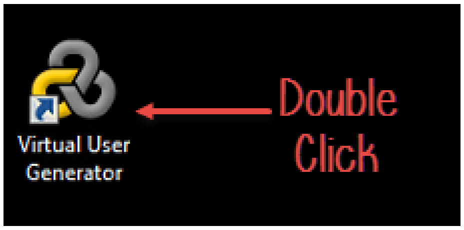
Web Tours

Welcome to the Web Tours site.
To make reservations,please enter your account information to the left.
If you haven't registered yet, **sign up now** to get access to all our resources
To configure the server options, use **administration** link.

This product uses parts of the SMT Kernel, Copyright (c) 1991-99 **iMatix Corporation**

Understanding the VUGen Environment

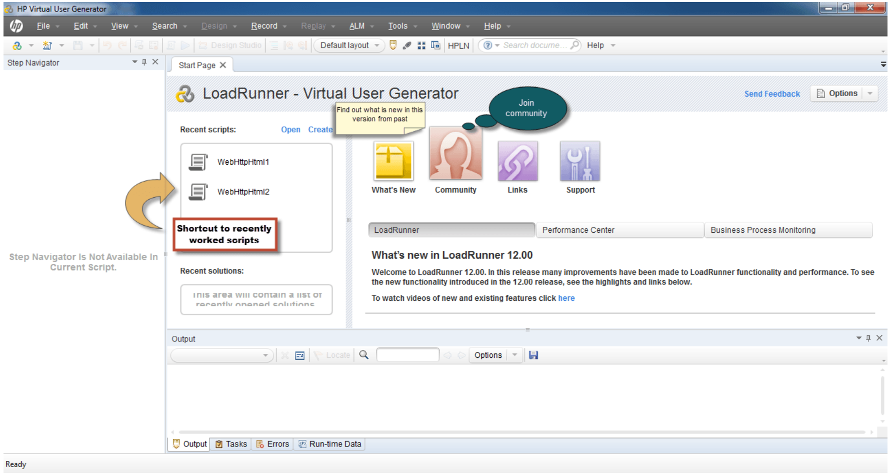
To run the VUGen, double click on the VUGen shortcut from the desktop, or go to Start Menu => HP Software => and then click on Virtual User Generator.



VUGen splash screen will appearas follows

HP LoadRunner
Virtual User Generator
Version 12.00
The Industry-Standard Load Testing Solution
This program is protected by US and international copyright laws as described in Help About.

The splash screen will subsequently open HP VUGen Home page, as shown below:

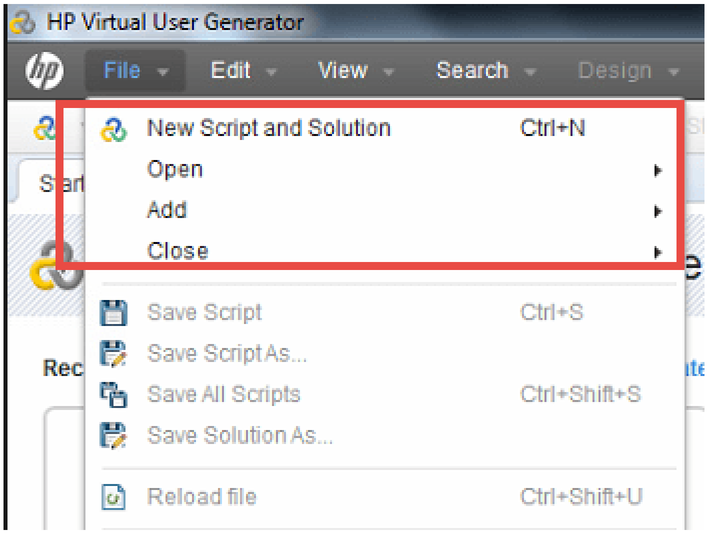


What's new in LoadRunner 12.00

Welcome to LoadRunner 12.00. In this release many improvements have been made to LoadRunner functionality and performance. To see the new functionality introduced in the 12.00 release, see the highlights and links below.

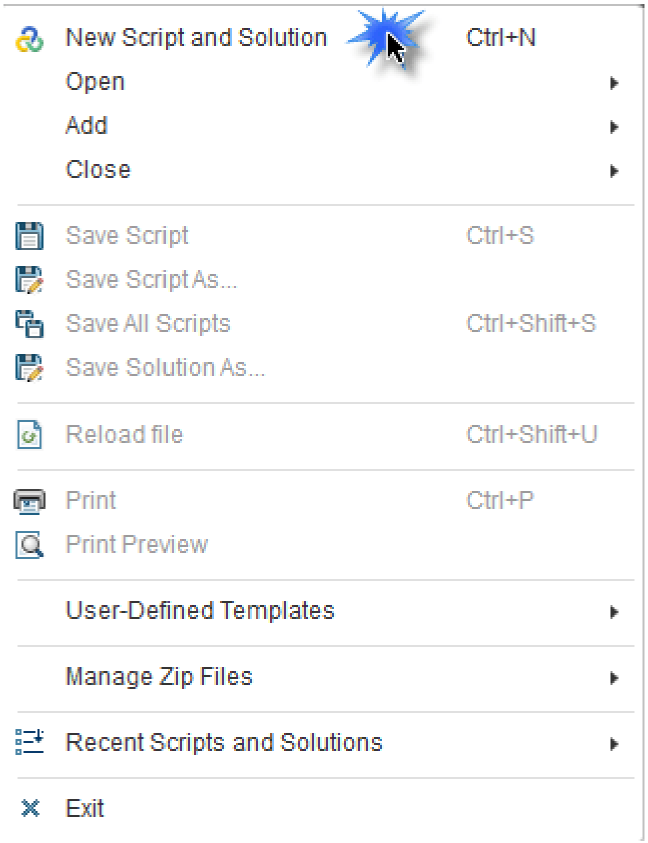
To watch videos of new and existing features click here

From the File menu, you can observe various command controls. The purpose of these controls are mentioned below:



New Script and Solution: is used to start creating a new script. Alternatively, you can create a new script from the toolbar button.
Open >> Script Solution: is used to open an existing, previously created script or solution from a directory.
Add >> New Script: is used to add a previously created script into current solution
We will get acquainted with other controls as we start creating our first script. Let's continue learning.

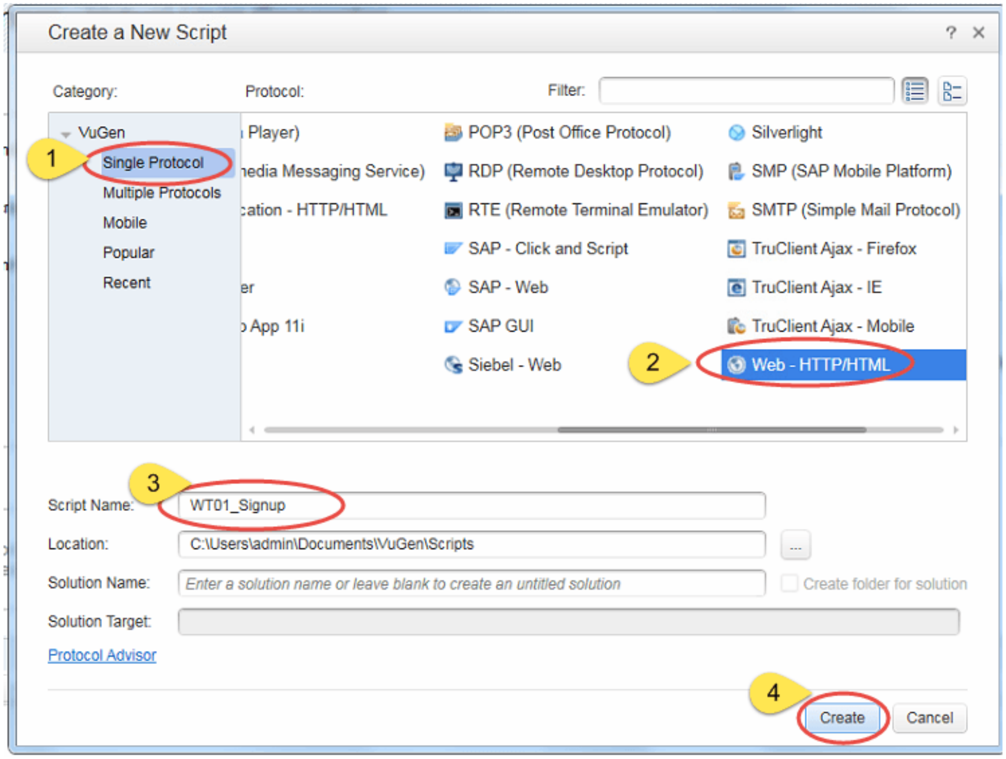
Creating Your First VUGen Script
Before you record, make sure the Web Tours App serveris running. Remember, you need to keep the server running so do not close it. If the window bothers you, you can minimize it.
Tip:In a real world problem, one needs to get acquainted with the subject application (SUL) especially if it involves complex business workflows and data stages. You can also interact with HP Web Tours to ensure you can repeat the exact steps when recording.
Before one can start with scripting of any application (SUL), one needs to get acquainted with the application flow. With this tutorial, let's get acquainted with HP Web Tours which is shipped and automatically installed with HP LoadRunner.
Step 1)Click on File =>New Script and Solution as shown in the snapshot below:



You can use the shortcut (Ctrl + N) to do the same.

Step 2) A window to select the protocol is shown. The list looks slightly different from previous versions

1. Select Single Protocol
2. Web
3. Enter Script Name
4. Click Create



Create a New Script

Category: VuGen > Single Protocol

Protocol: Web - HTTP/HTML

Script Name: WT01_Signup

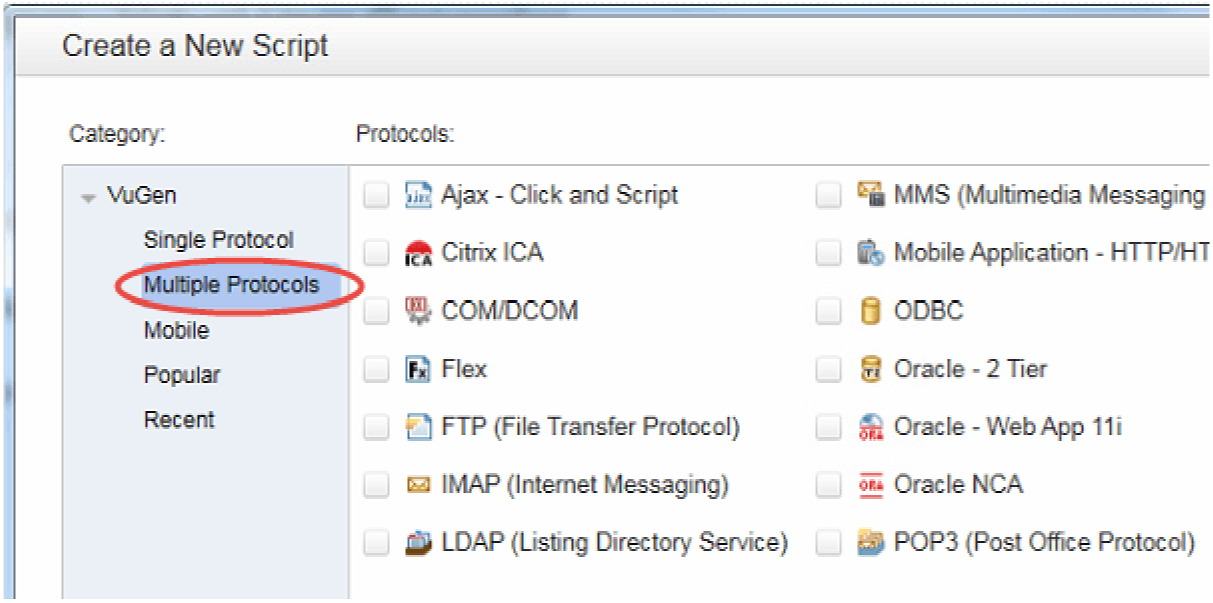
Location: C:\Users\admin\Documents\VuGen\Scripts

Solution Name: Enter a solution name or leave blank to create an untitled solution

Solution Target:

Note: You can use Protocol Advisor to determine right protocol your application uses for communication. For HP Web Tours, we already know that it requires “Web – HTTP/HTML” protocol. We will learn at a later stage how to determine the right protocol for your SUL.

Tip: An application may require more than 1 Protocols sometimes to communicate with its server. For such a case, navigate to Multiple Protocols list on the left side of window.



Create a New Script

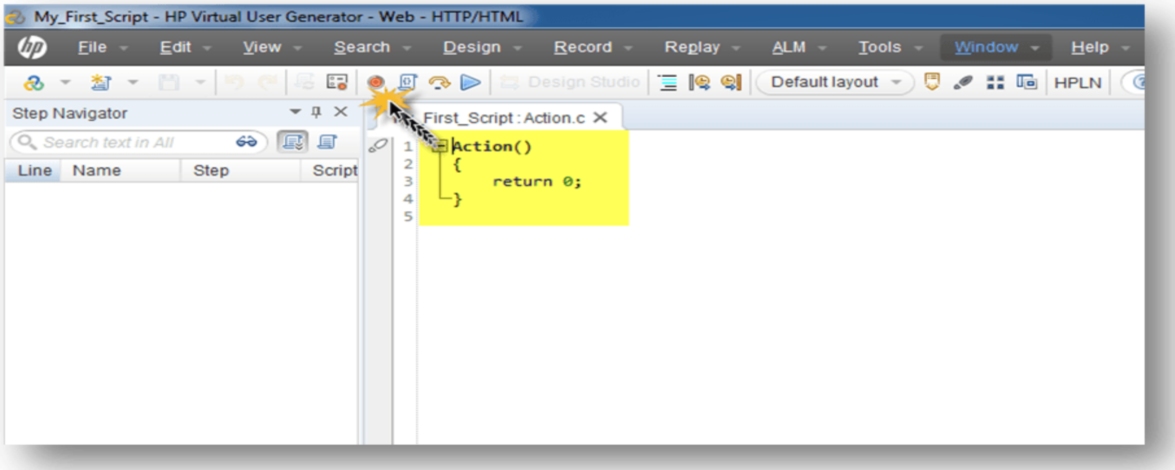
Category: VuGen > Multiple Protocols

Protocols:
VuGen
Single Protocol
Multiple Protocols
Mobile
Popular
Recent
□ Ajax - Click and Script
□ Citrix ICA
□ COM/DCOM
□ Flex
□ FTP (File Transfer Protocol)
□ IMAP (Internet Messaging)
□ LDAP (Listing Directory Service)
□ MMS (Multimedia Messaging
□ Mobile Application - HTTP/HT
□ ODBC
□ Oracle - 2 Tier
□ Oracle - Web App 11i
□ Oracle NCA
□ POP3 (Post Office Protocol)

HP Web Tours, however, requires only one protocol when scripting with VUGen.

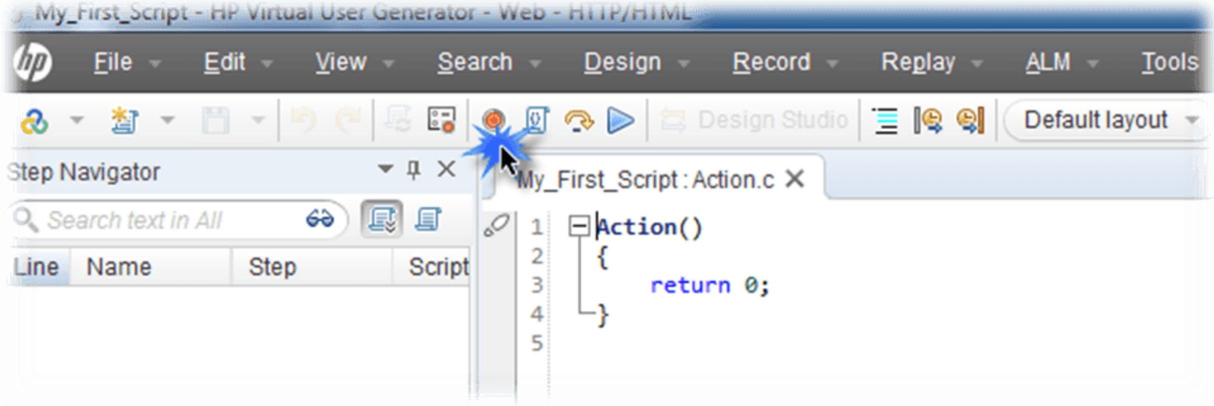
Tip: You cannot rename your scripts easily, so be creative when giving your scripts a name. If you’ve to rename a script, use Save As feature and give a new name. A good nomenclature could be something like, WT01_Signup where WT is short form of the application name, 01 is a business process sequence, and Sign up refers to the business processes being scripted. Also note that you cannot use spaces in a script name.

Step 3) once you click Create, HP VUGen will open the IDE (Integrated Development Environment) or code editor. You can notice the script files are blank, except basic signature of Function Action. Below snapshot should help understand it more:



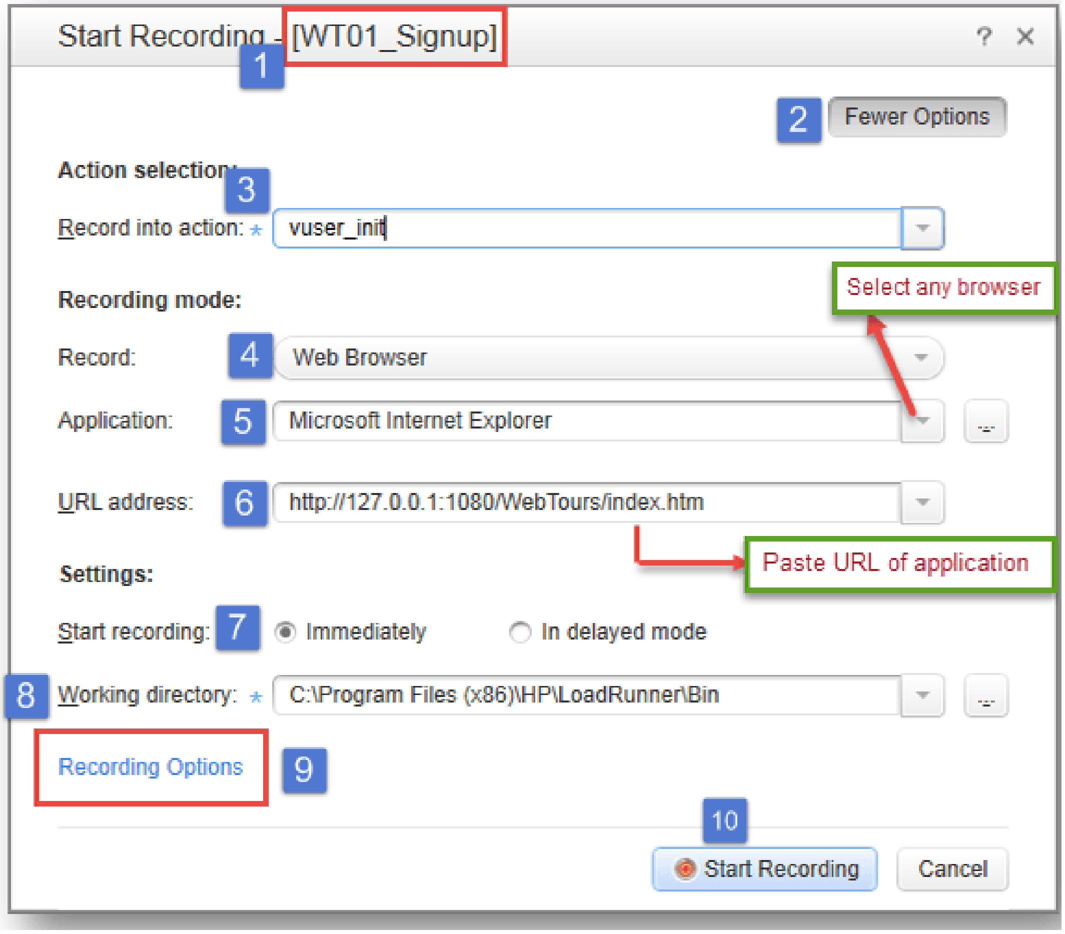
```
Action()
{
	return 0;
}
```

Step 4) Click the Recording button which you can find in the toolbar. Refer to the snapshot below:



```
Action()
{
	return 0;
}
```

Step 5) a new window opens



Start Recording - [WT01_Signup]

Action selection:

Record into action: vuser_init

Recording mode:

Record: Web Browser

Application: Microsoft Internet Explorer

URL address: http://127.0.0.1:1080/WebTours/index.htm

Settings:

Start recording: ● Immediately ○ In delayed mode

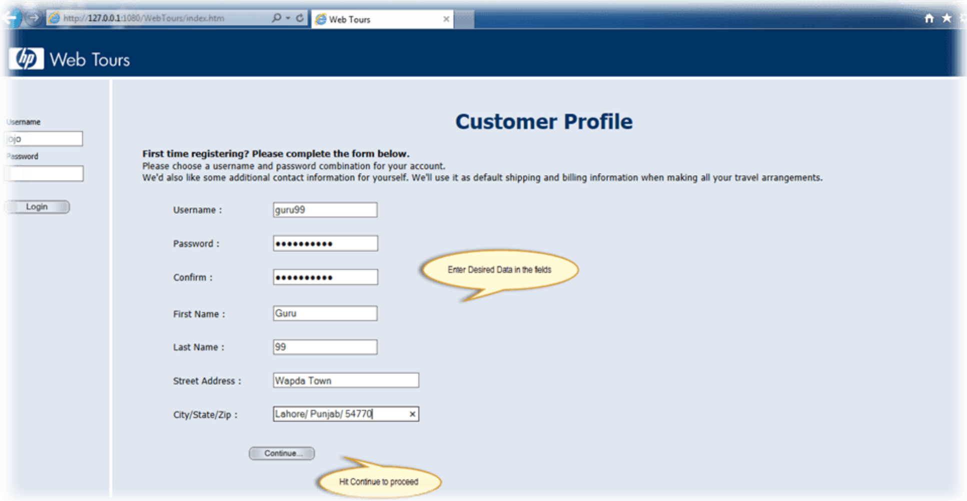
Working directory: C:\Program Files (x86)\HP\LoadRunner\Bin

1. If we study the above window, the name of the script is mentioned in the title bar of this popup. This will help you make changes to the right script when you are dealing with multiple scripts opened at a time.
2. If you do not see the rest of controls, click on the Fewer Options button to ensure you are not viewing fewer options.
3. Record into the action field determines where the code is generated after recording. VUGen facilitates switching to other actions or create new actions during and after recording.
4. Record: field tells VUGen that the subject application (SUL) is a web application or a desktop client. This option cannot be overridden once recording begins.
5. Application: field tells VUGen, which browser to use for recording. If you are not working with Web – HTTP/HTML protocol, you will select the desktop client with this field.

Tip: Unlike QTP scripts, LoadRunner scripts are independent of GUI. This is because the code generated does not work based on UI objects, rather, it works on the principle of client's request to the server and expecting server response back to the client – usually, in the form of HTML (hence the name of protocol)

6. URL Address: The URL of the SUL is input here. VUGen will invoke this URL when "Start Recording" is clicked. The browser used will be the one mentioned under "Record:" field if it is different from your default browser.
7. Start Recording: Immediate vs. Delayed: If you've selected Immediate recording mode, you will notice that VUGen starts recording events as soon as you hit "Start Recording" button. However, if you select Delayed recording mode, the VUGen will invoke the system (SUL) but will not commence capturing events unless you click on the Recording button from the floating bar.
8. Working Directory: This is a temporary directory which VUGen will use while capturing events. A copy of your script will be available at this location as well, just in case if you lose your folder :)
9.Recording Options: These settings tell VUGen what type of URLs to capture and what recording mode to use.
10.Start Recording: Will start recording. Click on it

Step 6) Once recording starts, VUGen will invoke the HP Web Tours application using the browser selected. Let’s Sign up a user at Web Tours application and record a basic script.


http://127.0.0.1:1080/WebTours/index.htm
Web Tours

Web Tours

Username
jojo
Password
Login

Customer Profile

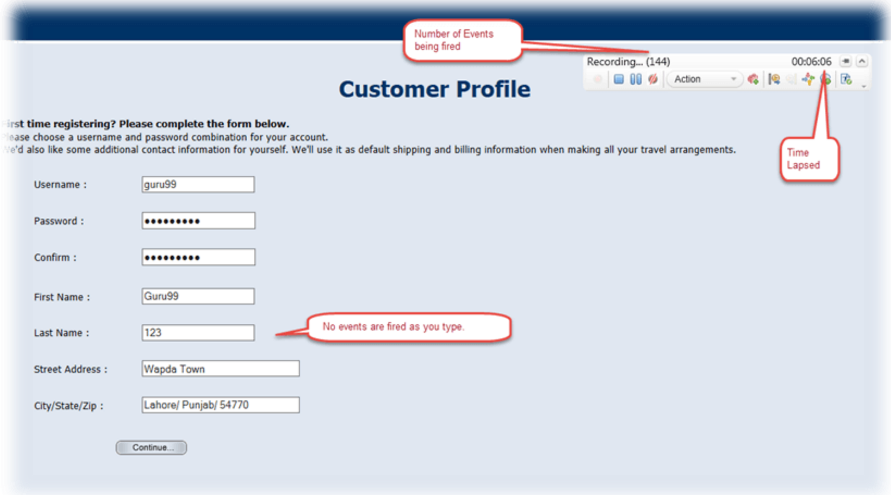
First time registering? Please complete the form below.
Please choose a username and password combination for your account.
We'd also like some additional contact information for yourself. We'll use it as default shipping and billing information when making all your travel arrangements.

Username : guru99
Password : ••••••••••
Confirm : ••••••••••
First Name : Guru
Last Name : 99
Street Address : Wapda Town
City/State/Zip : Lahore/ Punjab/ 54770

Continue...

Enter Desired Data in the fields

Hit Continue to proceed


Number of Events being fired

Recording... (144) 00:06:06
Action

Time Lapsed

Customer Profile

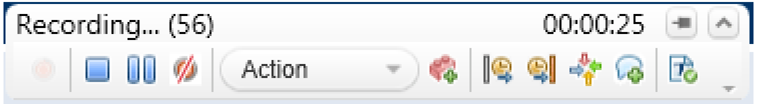
irst time registering? Please complete the form below.
ease choose a username and password combination for your account.
e'd also like some additional contact information for yourself. We'll use it as default shipping and billing information when making all your travel arrangements.

Username : guru99
Password : •••••••••
Confirm : •••••••••
First Name : Guru99
Last Name : 123
Street Address : Wapda Town
City/State/Zip : Lahore/ Punjab/ 54770

Continue...

No events are fired as you type.

A floating recording bar will appear which will offer various controls over recording. This tells the time elapsed and number of events automatically captured. Let’s look at the floating bar closely.


Recording... (56) 00:00:25
Action

In the snapshot shown above, the SUL has fired 56 events and a total of 0 minutes, and 25 seconds of time duration have elapsed since recording begun.

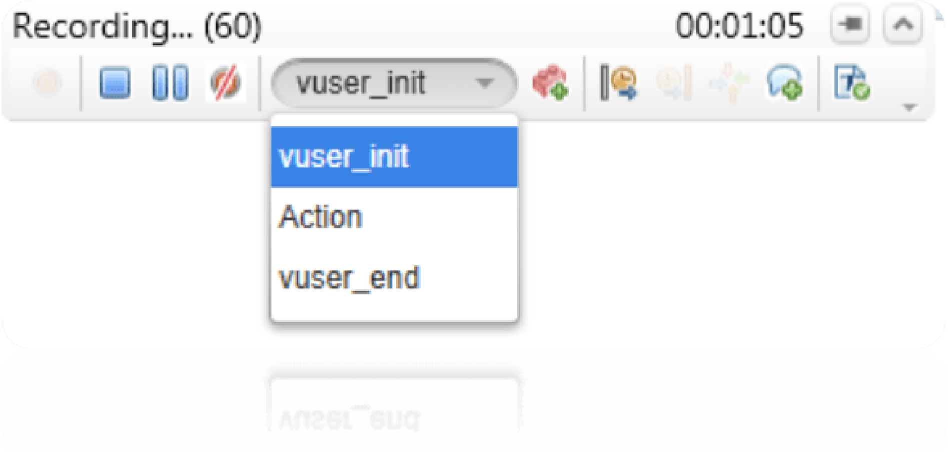
You’ll also observe that no events are fired as you type. Ask yourself why. Can you answer?

This is because no client to server communication takes place as you type. Hence, no events are fired and no script is generated regarding typing. Code will be generated only once you submit the form (click Continue)

Recording can be pause ⏸using the pause button. Please note, as long as the recording remains paused, all events being fired by the application will be disregarded. This also means that even if your application is sending requests and/or receiving response form SUL, the script will not be generated as long as you’ve paused the recording.

If you’ve paused recording, you will need to click recording ● button from the floating bar to resume recording.

While you are recording in vuser_init section, you’d notice that rendezvous button is not enabled. Switch to Action (or create a new Action) and continue recording. Refer to below snapshot to learn how to switch Action.



You'll notice that after switching to an Action, the rendezvous button will be enabled automatically. This is because VUGen does not allow insertion of rendezvous points in vuser_init.

Using Transactions at Record Time

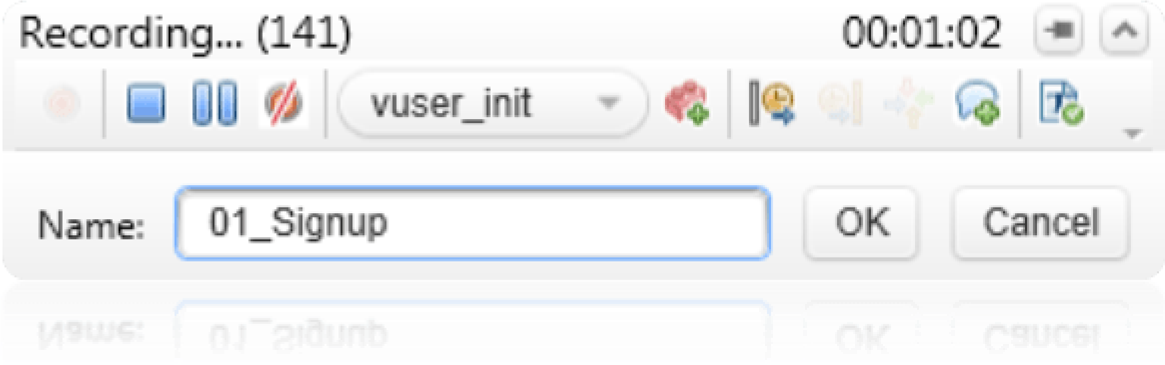
Let's assume you wish to gauge response time when X number of users simultaneously Sign up. To address this objective, let's read further.

By now you've recorded opening of the application in vuser_init action. After you've switched to Action, enter the user information required by SUL. Now before you hit "Continue" button, follow below steps:

Start a transaction by clicking button in floating bar. Enter the name of the transaction, for example, 01_Signup.

Tip: It is a good practice to name your transaction with a prefix like 01_ to help keep them unique.

The floating bar will look like this:

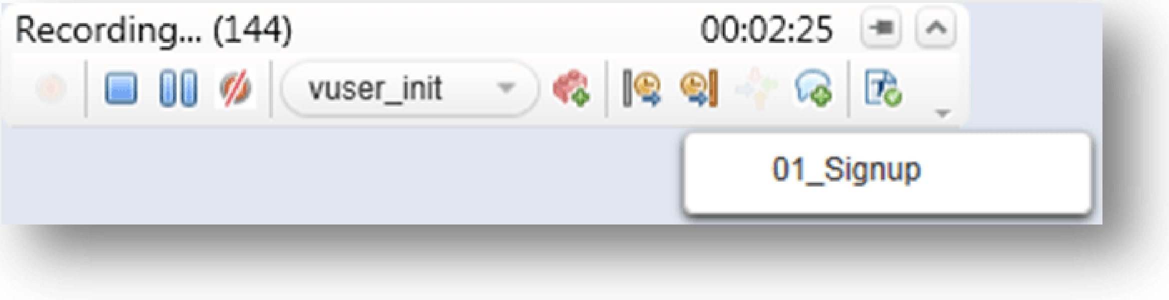


Name: 01_Signup

Click OK to close the transaction label.

Click the Continue button of the application.

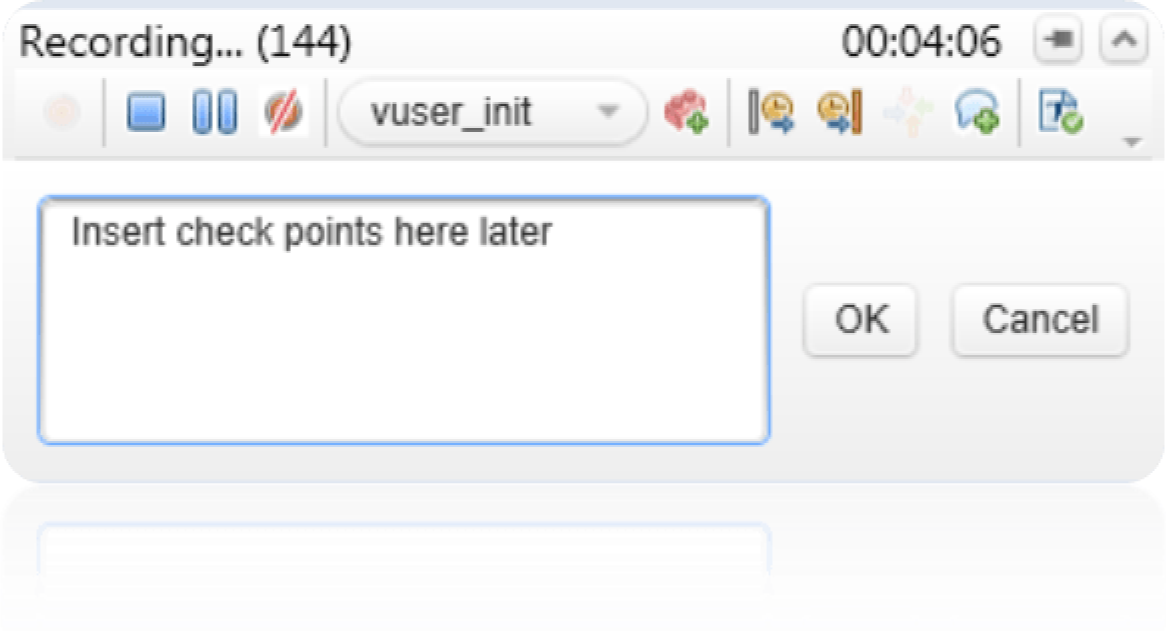
End transaction by clicking button. Select the name of the transaction you wish to close, for example, 01_Signup in this case. Refer to below snapshot for illustration.



If you do not have multiple transactions opened, you'll see only one name. If you've multiple, however, you'll 0be able to select from the list.

Insert Comments at Record Time:

Now that you've successfully closed the transaction, you can put a comment at record time to give yourself a reminder when you study the code later. From the floating bar, click on the button to insert a comment. A text box will appear adjacent to the floating bar where you can enter any text.



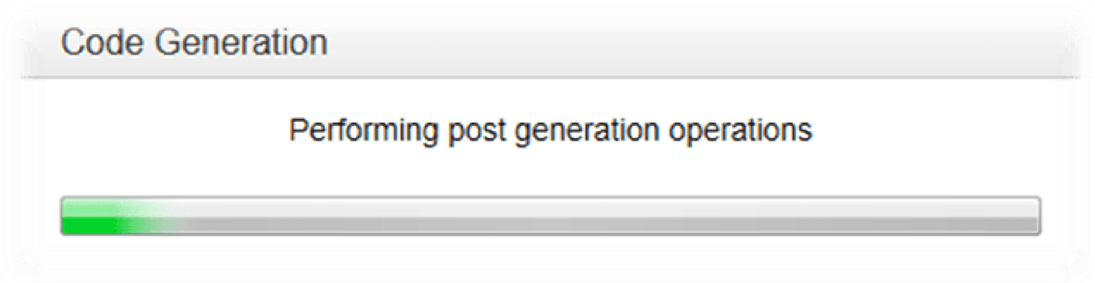
Insert check points here later

This comment will be inserted into the code after the steps (code) you've just finished before clicking OK button. From the application, you can click continue. You'll observe a welcome note by application.

Now select vuser_end action and click Sign off from the application. Close the application and click stop button 🟦 from the floating bar.

Code Generation:

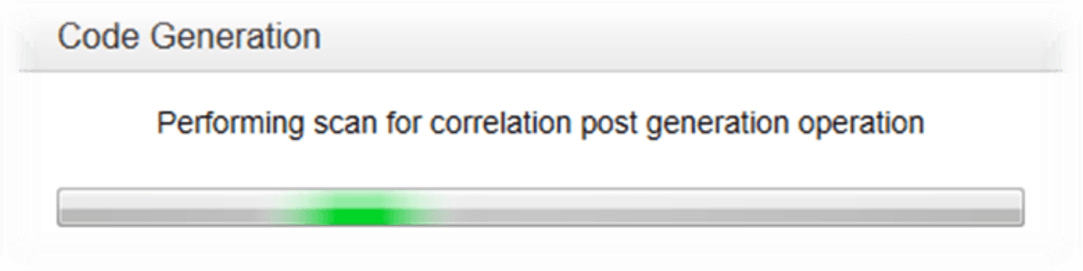
Once recording has been stopped, you'll notice VUGen post generation activity. This includes generating code, transactions and comments based on your recording. Refer to snapshot below.



Code Generation

Performing post generation operations

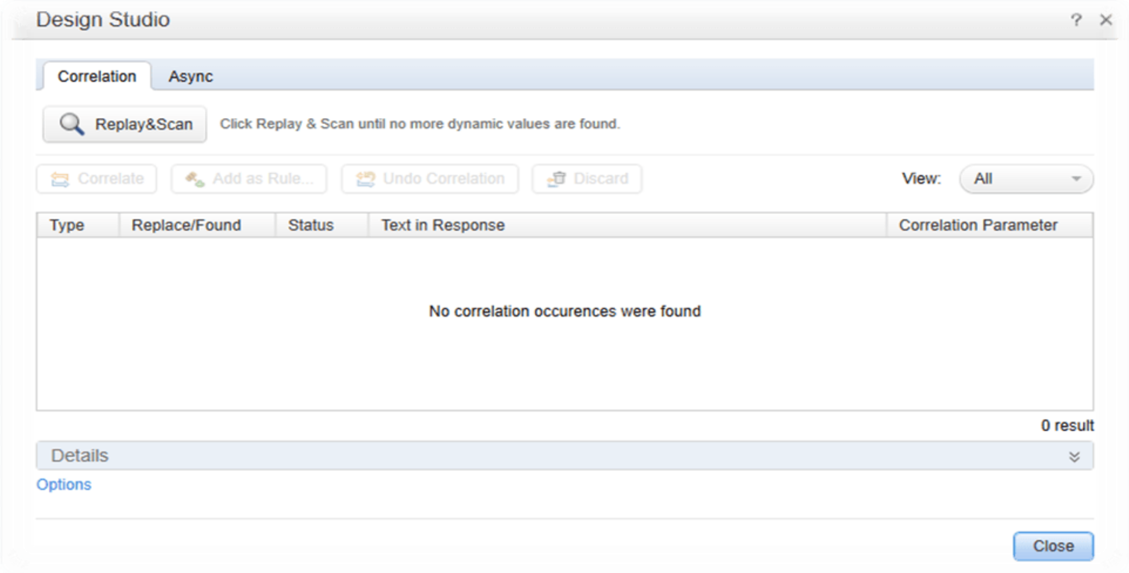
Immediately after VUGen finished "Performing post generation operations" it will start "Performing scan for correlation post generation operation" We will see in more detail what correlation means.



Code Generation

Performing scan for correlation post generation operation

Once post generation operation is finished, you will see Design Studio window. If any candidates for correlation are found, you'll be able to review them in this window.



Design Studio

Click Replay & Scan until no more dynamic values are found.

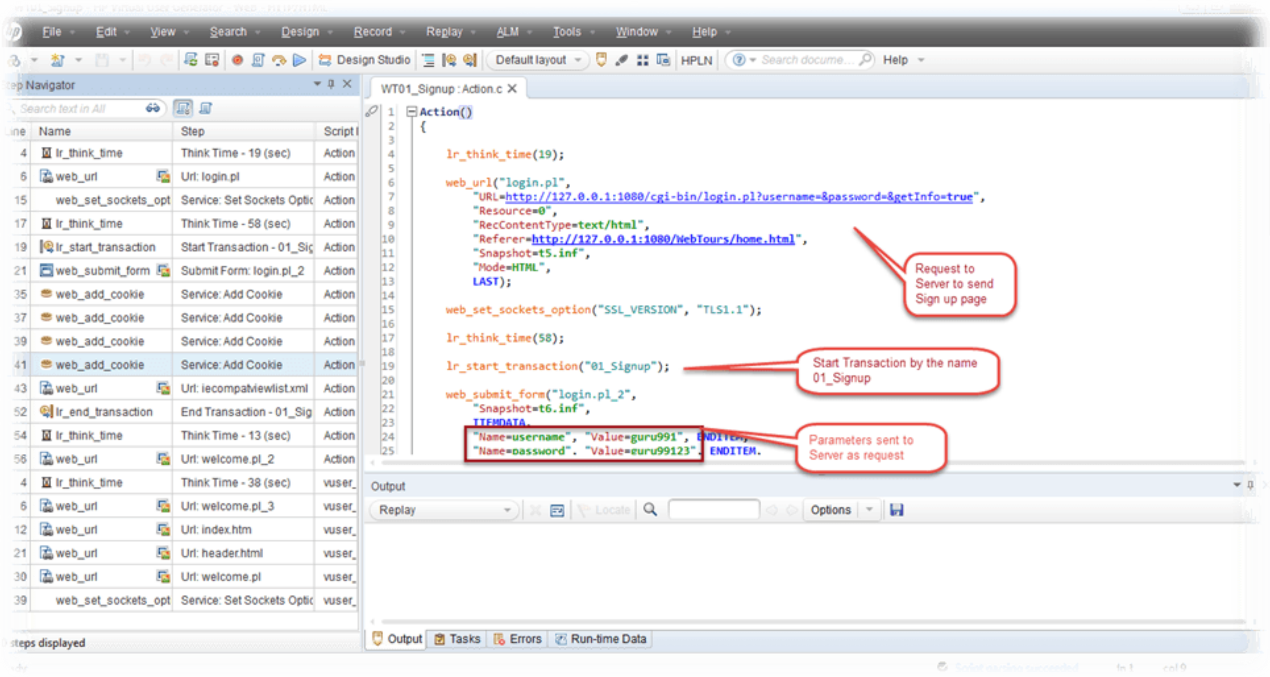
View: All

Type	Replace/Found	Status	Text in Response	Correlation Parameter

No correlation occurences were found

0 result

You can safely click on the Close button, and VUGen will take you to the code that has been generated. The window will look like this:



Line	Name	Step	Script
4	lr_think_time	Think Time - 19 (sec)	Action
6	web_url	Url: login.pl	Action
15	web_set_sockets_opt	Service: Set Sockets Optic	Action
17	lr_think_time	Think Time - 58 (sec)	Action
19	lr_start_transaction	Start Transaction - 01_Sig	Action
21	web_submit_form	Submit Form: login.pl_2	Action
35	web_add_cookie	Service: Add Cookie	Action
37	web_add_cookie	Service: Add Cookie	Action
39	web_add_cookie	Service: Add Cookie	Action
41	web_add_cookie	Service: Add Cookie	Action
43	web_url	Url: iecompatviewlist.xml	Action
52	lr_end_transaction	End Transaction - 01_Sig	Action
54	lr_think_time	Think Time - 13 (sec)	Action
56	web_url	Url: welcome.pl_2	Action
4	lr_think_time	Think Time - 38 (sec)	vuser_
6	web_url	Url: welcome.pl_3	vuser_
12	web_url	Url: index.htm	vuser_
21	web_url	Url: header.html	vuser_
30	web_url	Url: welcome.pl	vuser_
39	web_set_sockets_opt	Service: Set Sockets Optic	vuser_

steps displayed

```
Action()
{

    lr_think_time(19);

    web_url("login.pl",
        "URL=http://127.0.0.1:1080/cgi-bin/login.pl?username=&password=&getInfo=true",
        "Resource=0",
        "RecContentType=text/html",
        "Referer=http://127.0.0.1:1080/WebTours/home.html",
        "Snapshot=t5.inf",
        "Mode=HTML",
        LAST);

    web_set_sockets_option("SSL_VERSION", "TLS1.1");

    lr_think_time(58);

    lr_start_transaction("01_Signup");

    web_submit_form("login.pl_2",
        "Snapshot=t6.inf",
        ITEMDATA,
        "Name=username", "Value=guru991", ENDITEM,
        "Name=password", "Value=guru99123", ENDITEM,
```

Request to Server to send Sign up page

Start Transaction by the name 01_Signup

Parameters sent to Server as request

As highlighted in the above figure, the editor contains easy to understand, readable code. You can have a look on the parameters used while recording.

The left side of VUGen is called Step Navigator that lets you understand the “script” without looking at the granularities of the code. For example, you can read steps as, opening a URL, spend Think Time and submit the form. This encapsulates all the parameters associated with each request.

Your script has been generated, click on the File menu and then click on Save Script As to browser the location where you wish to save your script. VUGen will automatically suggest the last directory used if you’ve already saved a script earlier. The name suggested will be the name you mentioned when you started recording.

All done.

Now, congratulate yourself for having your first script successfully generated and saved.

Deciding a Protocol and Protocol Advisor

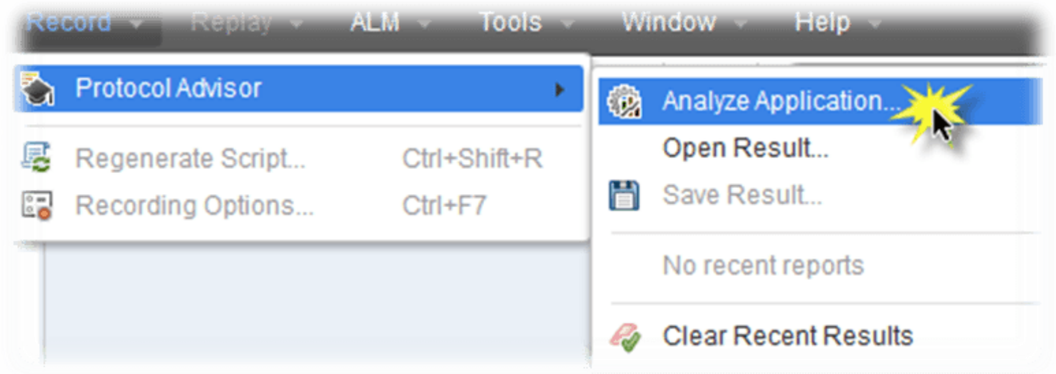
You might have questioned yourself why did we use Web – HTTP/HTML protocol. How we decided which protocol should we use? The answer is not that simple.

There is an architectural foundation set of skills you need to have in place as a prerequisite to answering this question. If you are a beginner, you can pair up with someone who has solid client-side architectural and development skills for your SUL. You can speak with the developers of your SUL and find out which interfaces your application leverages. This should lead you on a natural path to the interfaces that you will be using for your virtual user script development and protocol selection.

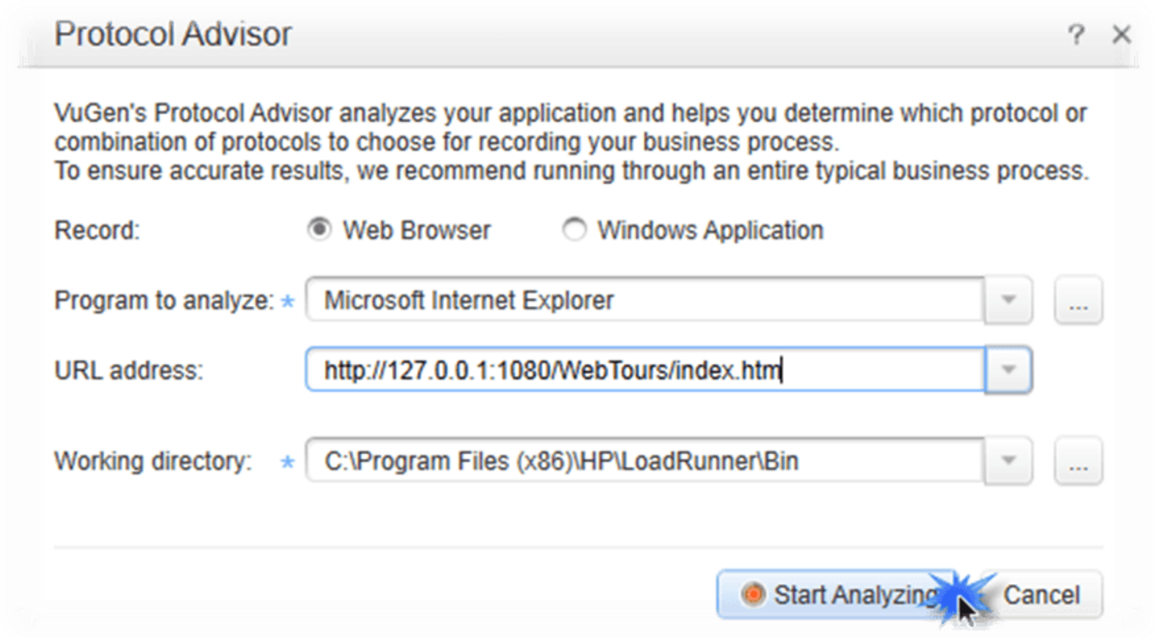
To address the needs of beginners less knowledgeable in architectural skills, LoadRunner introduced a feature called Protocol Advisor in LoadRunner 9.1. While this has made life easier for many, one should rely more on the architectural and development skills instead of protocol advisor and receiving information from the development team about underlying development technologies. Protocol may not suggest correct protocol in all cases.

To use Protocol Advisor, go to Record => Protocol Advisor => Analyze Application

Refer to snapshot below:



This will open the main window of Protocol Advisor. If you notice, this resembles a bit with the window appearing for recording. Let’s have a look at the window below:



VuGen's Protocol Advisor analyzes your application and helps you determine which protocol or combination of protocols to choose for recording your business process.
To ensure accurate results, we recommend running through an entire typical business process.

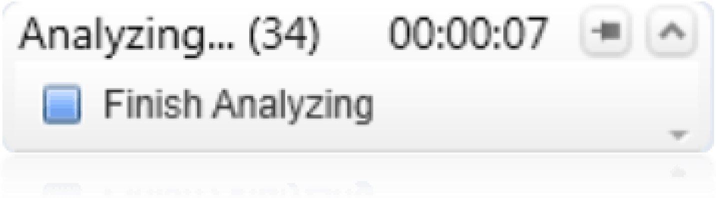
Record: ● Web Browser ○ Windows Application

Program to analyze: * Microsoft Internet Explorer

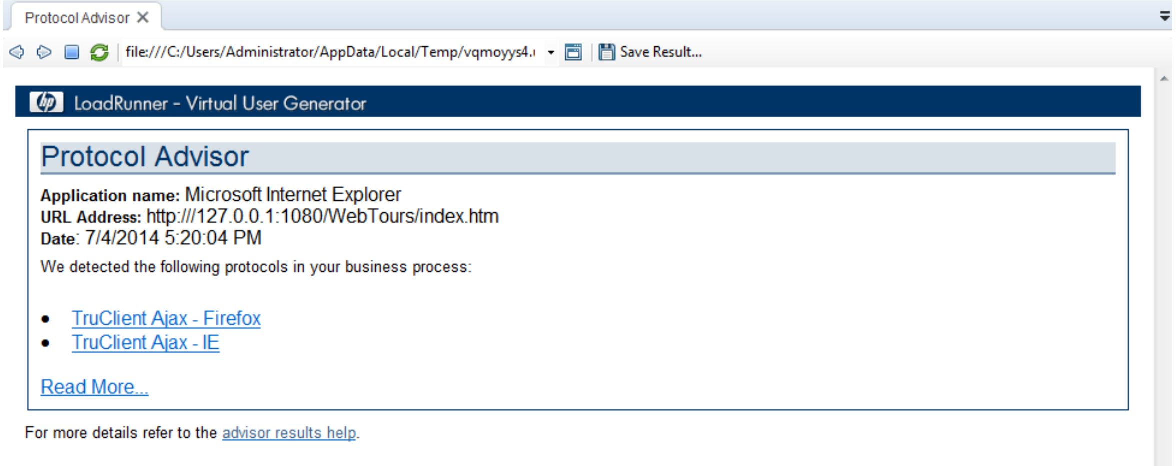
URL address: http://127.0.0.1:1080/WebTours/index.htm

Working directory: * C:\Program Files (x86)\HP\LoadRunner\Bin

Select the Web Browser since we are using a web based application.
Specify the URL of the application that will subsequently be invoked. Working directory can be left as such since this is merely a temporary directory for VUGen to use. Ensure you've read and write access on this directory.
Click the Start Analyzing button.
A floating bar, somewhat similar to the record time floating bar will appear. Have a look at the snapshot:


Analyzing... (34) 00:00:07

The process will tell the time elapsed and number of events fired. However, this information is not necessary. The only benefit of this events counter is, you know your client, SUL, is communicating with the server.
It is a good practice to analyze only one business process at a time since various business process in a large enterprise application may involve various protocols. For example, a dashboard in some application may have Ajax or Flex, etc. but this will not be present on the login page.
Once you've finished executing a particular business process, you can hit the Stop button. The VUGen protocol advisor will come up with a summary report on the protocol suggestion. Have a look how it looks like:


Protocol Advisor ×
file:///C:/Users/Administrator/AppData/Local/Temp/vqmoyys4.
Save Result...
LoadRunner - Virtual User Generator

Protocol Advisor

Application name: Microsoft Internet Explorer
URL Address: http:///127.0.0.1:1080/WebTours/index.htm
Date: 7/4/2014 5:20:04 PM

We detected the following protocols in your business process:

- TruClient Ajax - Firefox
- TruClient Ajax - IE

Read More...

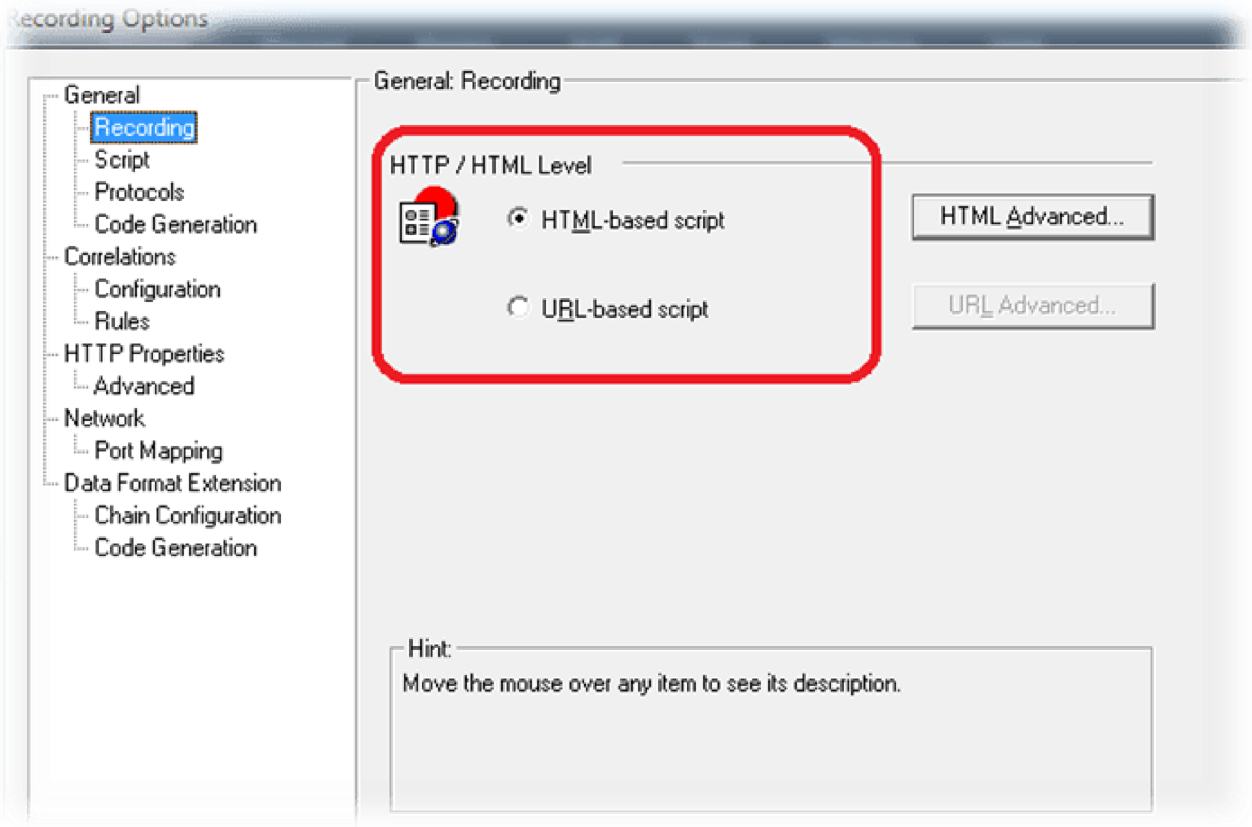
For more details refer to the advisor results help.

You can see the suggestions from Protocol Advisor. These may or may not be the best choices to pick.
You've learned to use Protocol Advisor by now. However, this could be help for beginners or for situation where you need "another opinion" – rely on your architectural sense, programming knowledge, development skills and information received from development team to decide on the protocol.

Recording Options

Whenever VUGen generates a script, the code generated is based on various configurations that can be found under the "Recording Options" – or you can press Ctrl + F7 to view the Recording Options.

Let's have a look at recording options window before we discuss all configurations:



● HTML-based script
○ URL-based script

Move the mouse over any item to see its description.

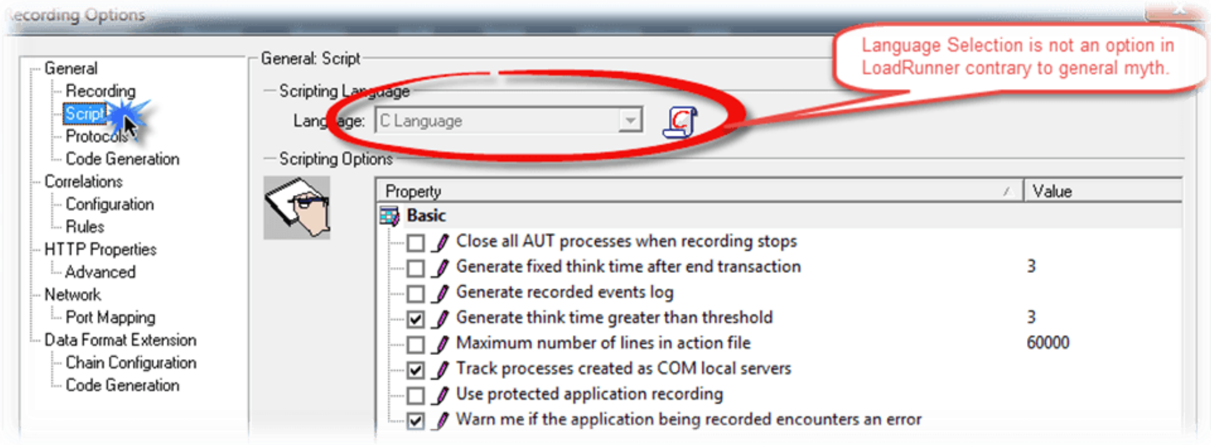
There are various categories of configurations like General, Correlations, Network and Data Format Extension. Let's understand most significant among these, one by one.

General => Recording:

This topic requires detail understanding. Hence this is discussed separately.

General => Script:

Have a look at the snapshot for a glimpse:



Language Selection is not an option in LoadRunner contrary to general myth.

General: Script

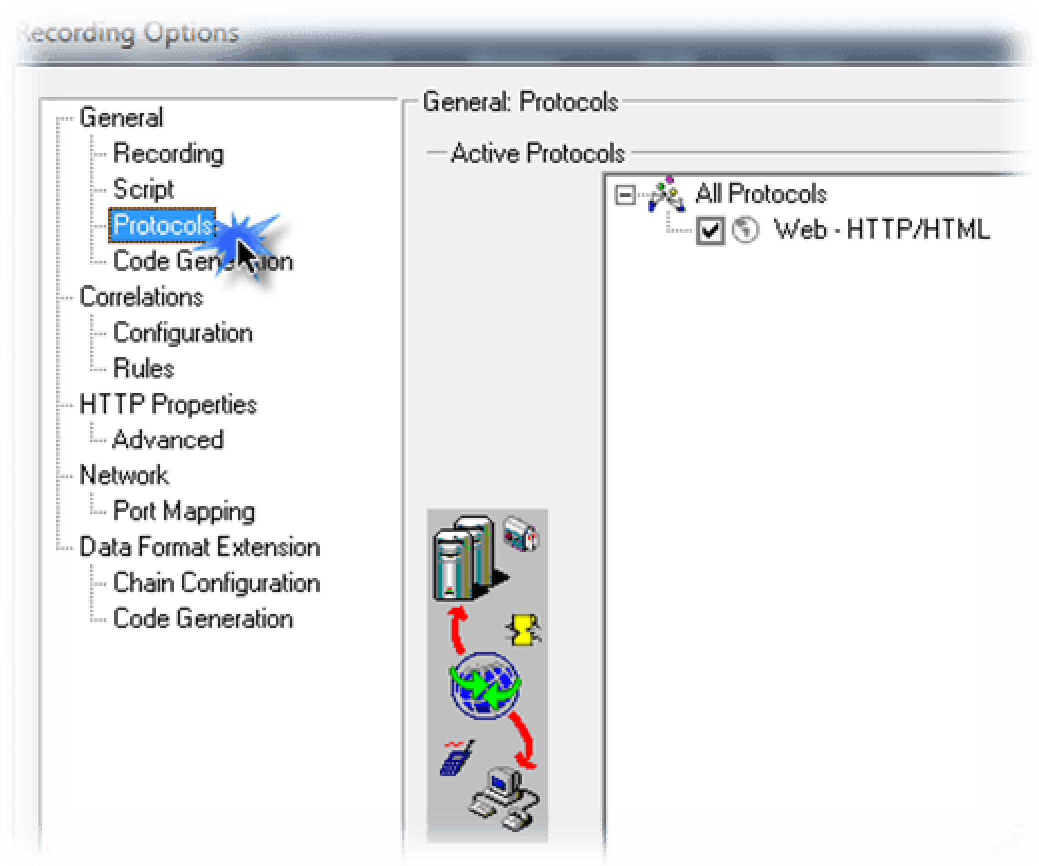
Scripting Language

Language: C Language

Scripting Options

Property	Value
Basic	
□ Close all AUT processes when recording stops	
□ Generate fixed think time after end transaction	3
□ Generate recorded events log	
☑ Generate think time greater than threshold	3
□ Maximum number of lines in action file	60000
☑ Track processes created as COM local servers	
□ Use protected application recording	
☑ Warn me if the application being recorded encounters an error	

You'll notice that Language dropdown is disabled. A common myth is that the LoadRunner does not generate code in any other language. Another myth is that it requires a license to work in other languages.
Both are false. LoadRunner decides for itself which language to use when generating the script. In almost all cases, you'll find yourself working with C Language.
For certain java applications (like Java applets) the code being generated will be in Java Script Language.
VUGen will generate a script in VBScript Language only for applications developed in Visual Basic classic (MS Visual Studio 2002)
Scripting Options:
You can opt to "Generate fixed think time after end transaction". This means, no matter how much a user wait, the think time generated (the delay) will be equal to value specified. The value is in seconds.
Maximum number of lines in the action file refers to the maximum number of lines VUGen will generate in an action. If the script is larger, VUGen will automatically create a new action. The default is set to 60,000. The maximum value which can be specified is 65,000
You may find this configuration helpful when dealing with a desktop application with Oracle on the backend.
General => Protocol gives you an option to select and deselect any protocols you've selected at the start of recordingp
Essentially, this will be used only when you wish to Re-Generate Script.
Have a look at the screen:

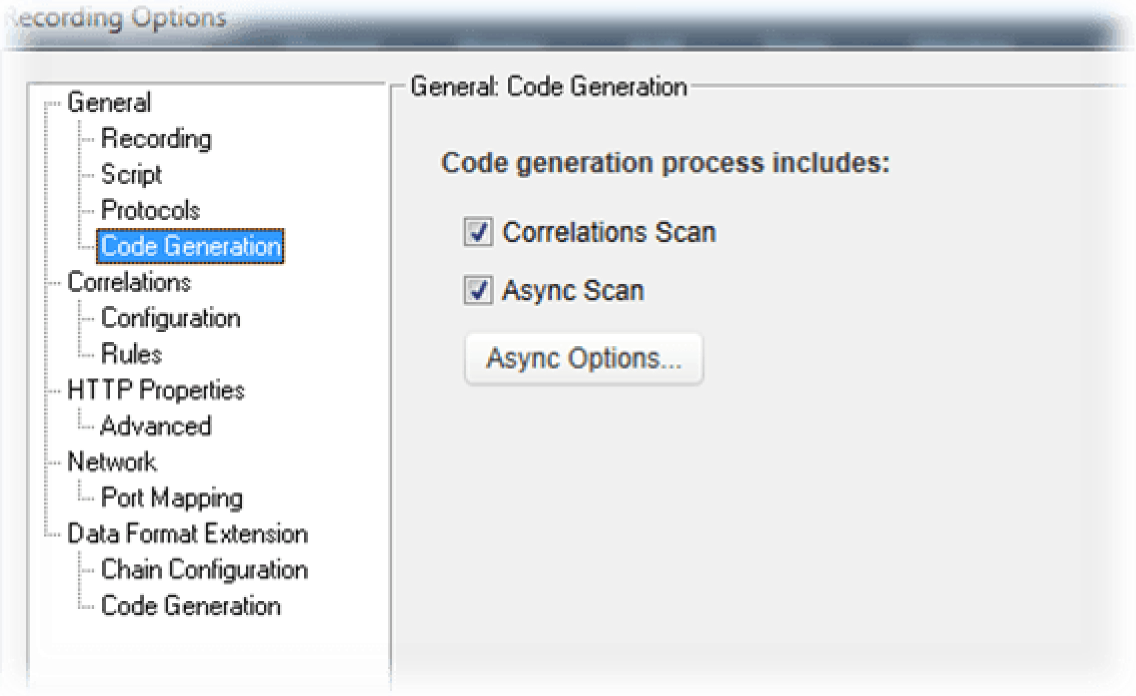


☑ Web - HTTP/HTML

This is helpful when you've used multi protocols at the time of recording a script. You can regenerate the script and deselect the protocols you don't wish and get a new script without having to re-record it.

General => Code Generation:

Have a look at the snapshot below:



Code generation process includes:

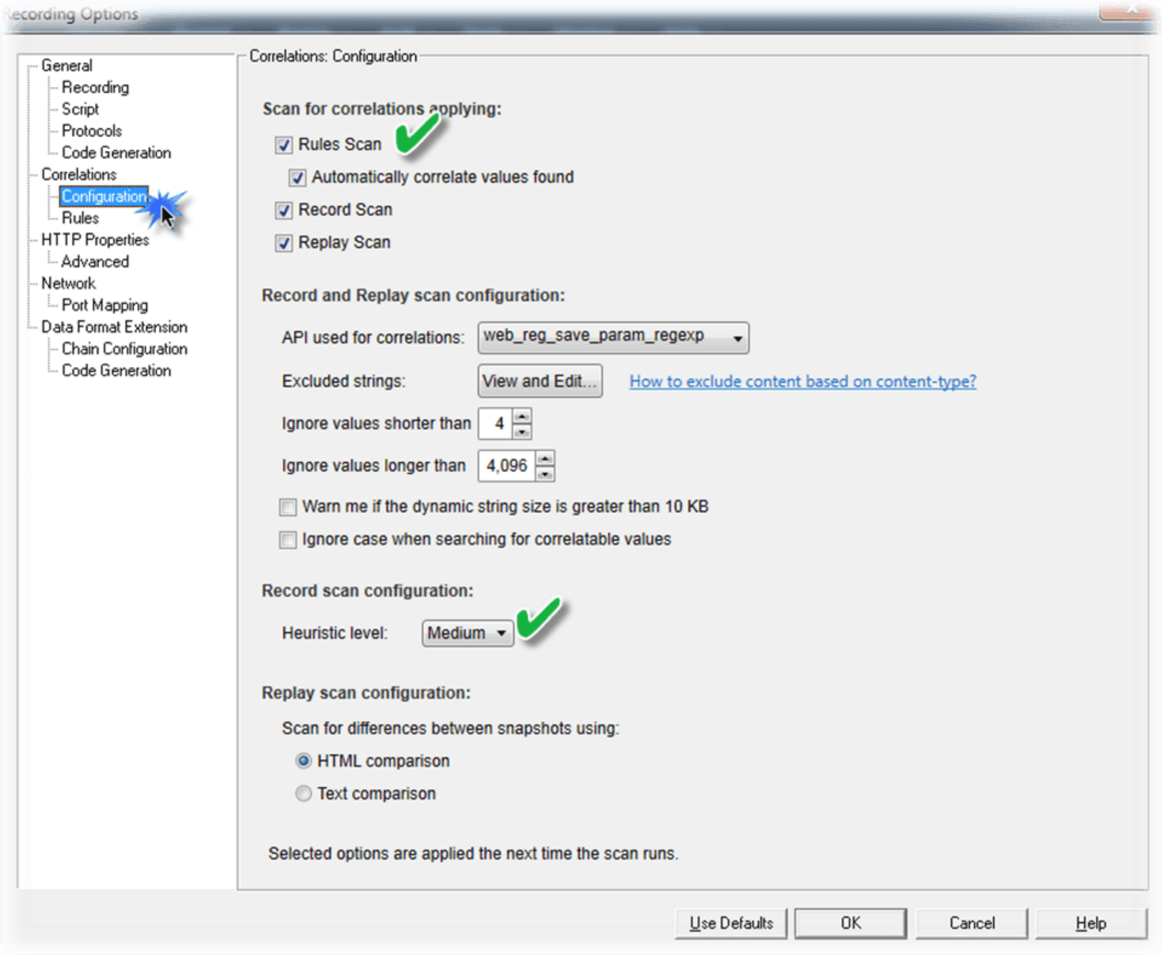
☑ Correlations Scan

☑ Async Scan

This configuration tells VUGen to find candidates for correlation at record time. If you do not wish for Automatic Correlation, then you might wish to turn off this feature.

Correlation => Configuration:

Have a look at the screenshot below and familiarize yourself with the screen.



Correlations: Configuration

Scan for correlations applying:

☑ Rules Scan

☑ Automatically correlate values found

☑ Record Scan

☑ Replay Scan

Record and Replay scan configuration:

API used for correlations: web_reg_save_param_regexp

Excluded strings: View and Edit... How to exclude content based on content-type?

Ignore values shorter than 4

Ignore values longer than 4,096

□ Warn me if the dynamic string size is greater than 10 KB

□ Ignore case when searching for correlatable values

Record scan configuration:

Heuristic level: Medium

Replay scan configuration:

Scan for differences between snapshots using:

● HTML comparison

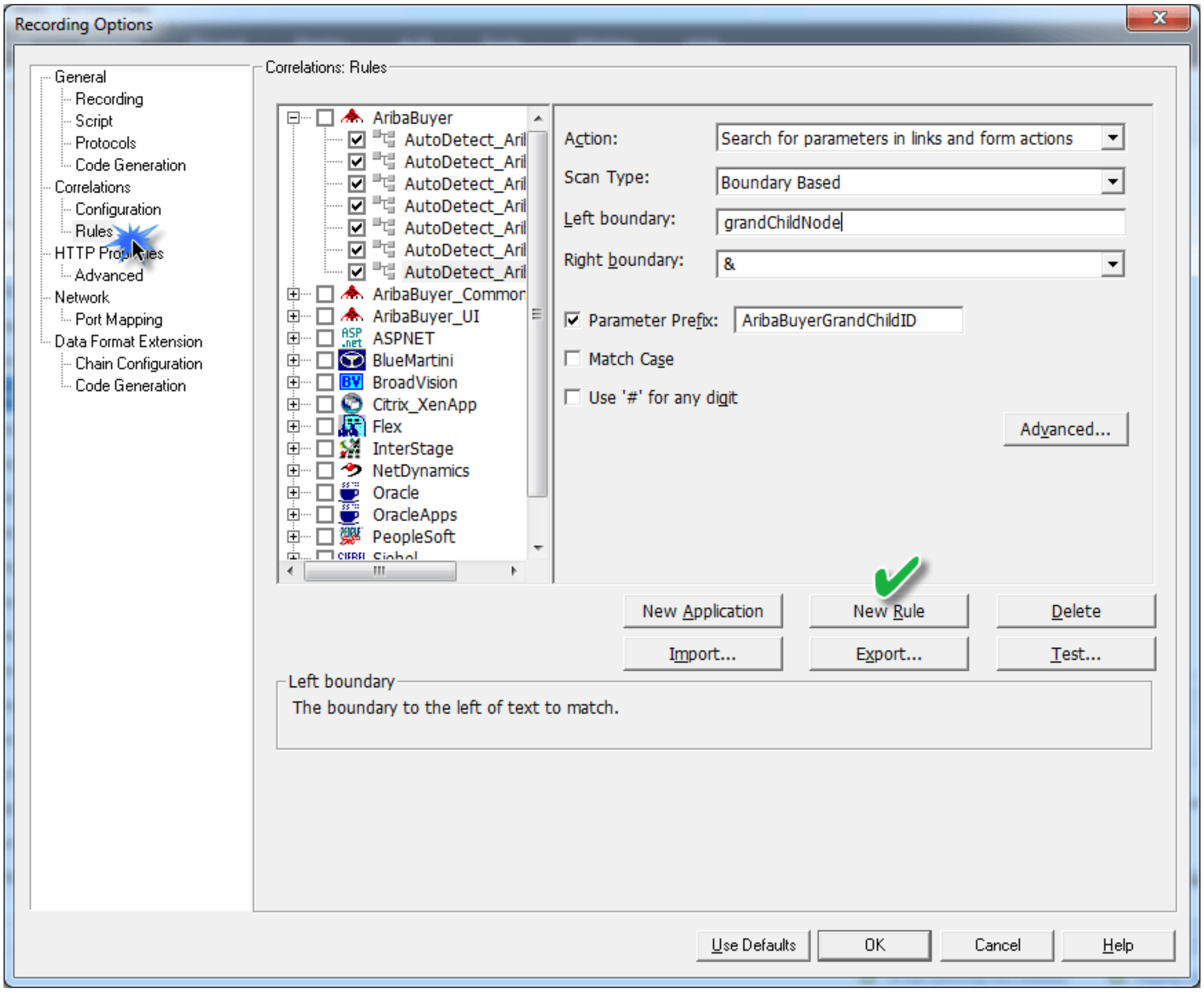
○ Text comparison

Selected options are applied the next time the scan runs.

Although automatic correlation is helpful from 5% to 10% only, yet you can select “Rules Scan” and “Automatically correlate values found”. However, if your script doesn’t play, you can consider restoring to defaults by clicking on button.

Correlation => Rules:

Go to Rules, and here you can see various rules VUGen is using to find correlation candidates. You can add custom rules if you know what your application (SUL) is using as parameters. However, this is an advanced use of record time settings. If you’re a beginner, you can safely skip this topic.



Action: Search for parameters in links and form actions

Scan Type: Boundary Based

Left boundary: grandChildNode

Right boundary: &

☑ Parameter Prefix: AribaBuyerGrandChildID

□ Match Case

□ Use '#' for any digit

Left boundary

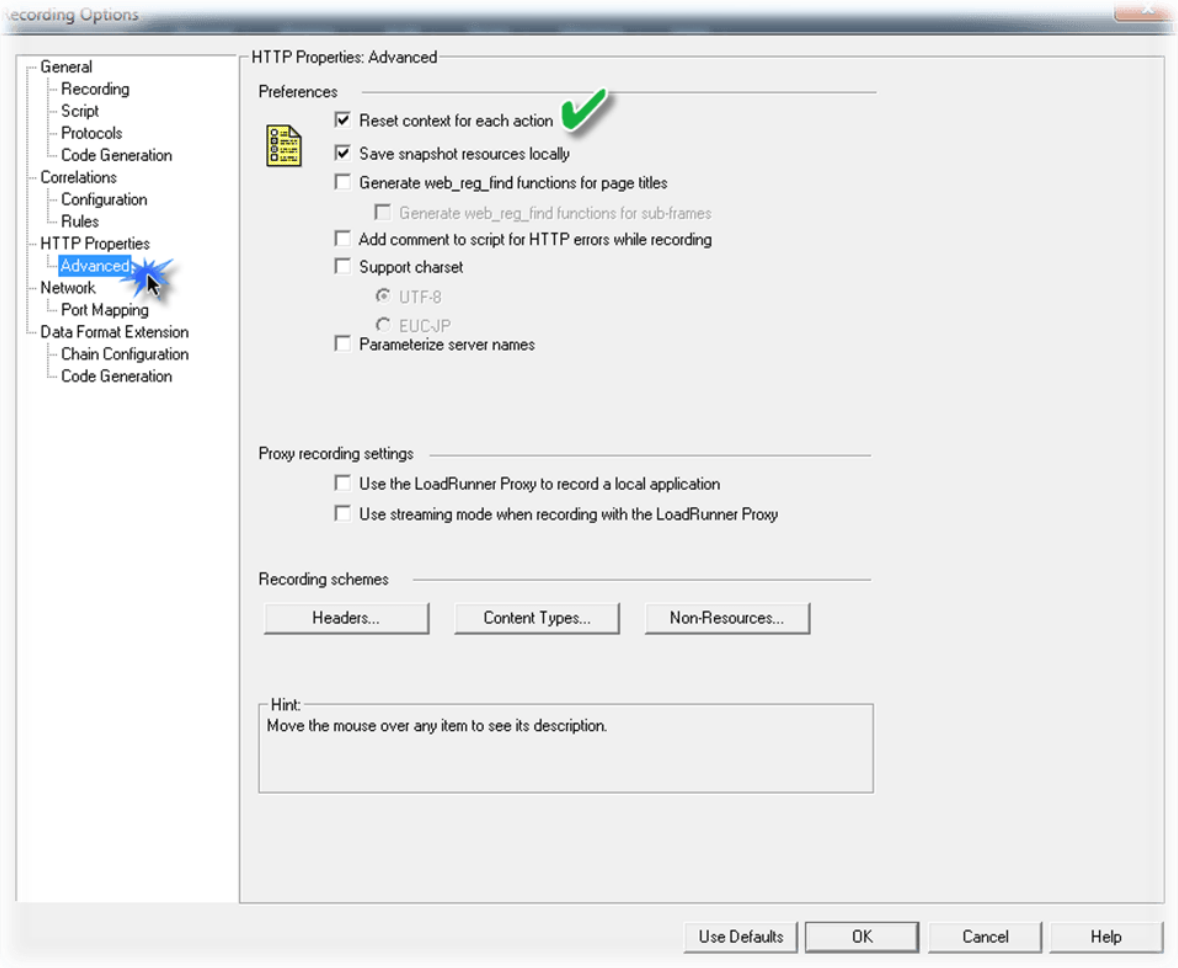
The boundary to the left of text to match.

HTTP Properties => Advanced:

This frame offers various settings related to HTTP binding.

Reset context for each action, enabling this option instructs VUGen to reset all HTP contexts between actions to their initial state before recording, providing a clean beginning for the recording session. The option is enabled by default.

You can leave the rest of configurations intact, unless required.



HTTP Properties: Advanced

Preferences

☑ Reset context for each action

☑ Save snapshot resources locally

□ Generate web_reg_find functions for page titles

□ Generate web_reg_find functions for sub-frames

□ Add comment to script for HTTP errors while recording

□ Support charset

● UTF-8

○ EUC-JP

□ Parameterize server names

Proxy recording settings

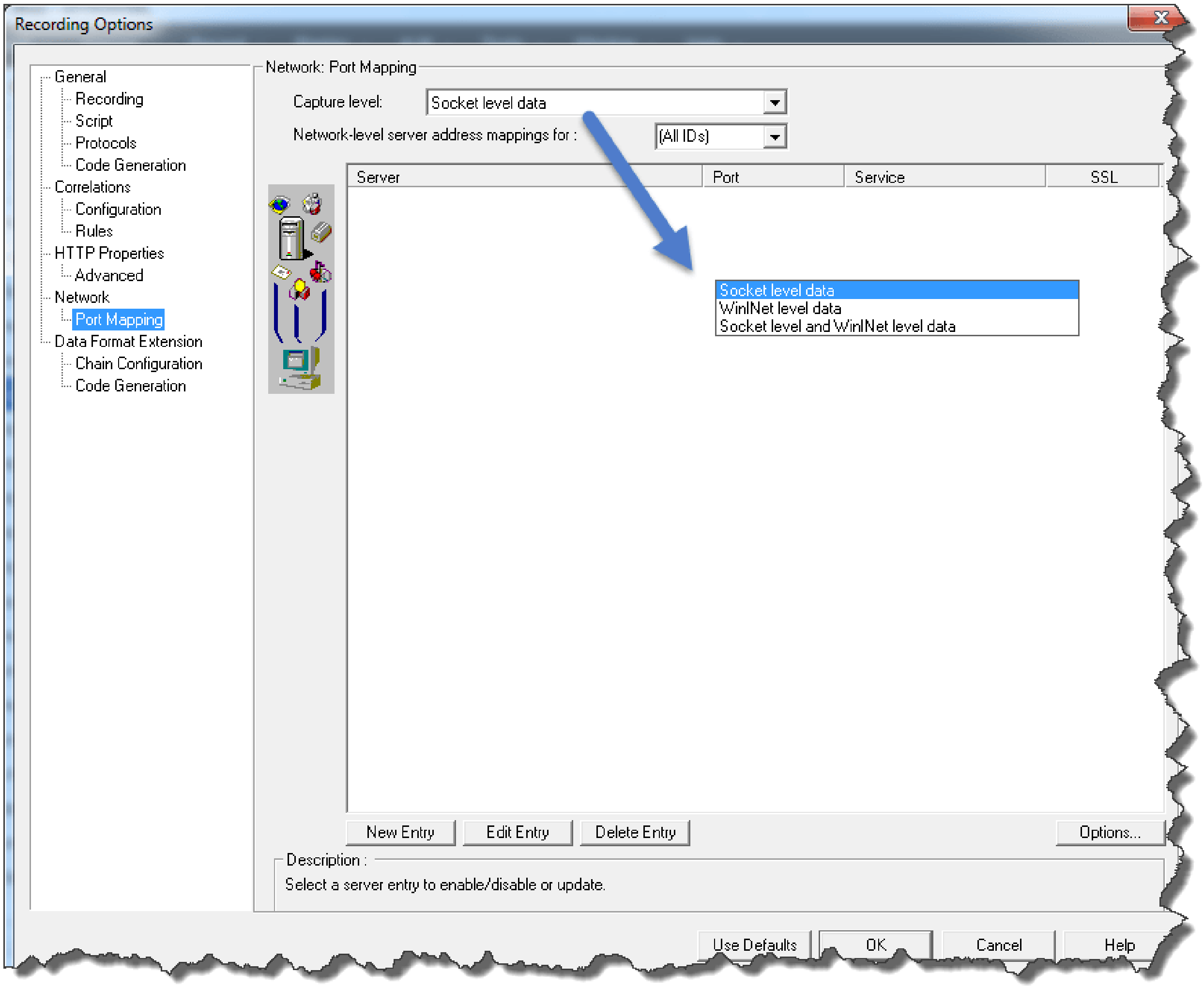
□ Use the LoadRunner Proxy to record a local application

□ Use streaming mode when recording with the LoadRunner Proxy

Recording schemes

Hint:
Move the mouse over any item to see its description.

Network => Port Mapping:
This frame should be left intact. If you’re recording a desktop application, then you may have to choose WinINet level data.



Network: Port Mapping

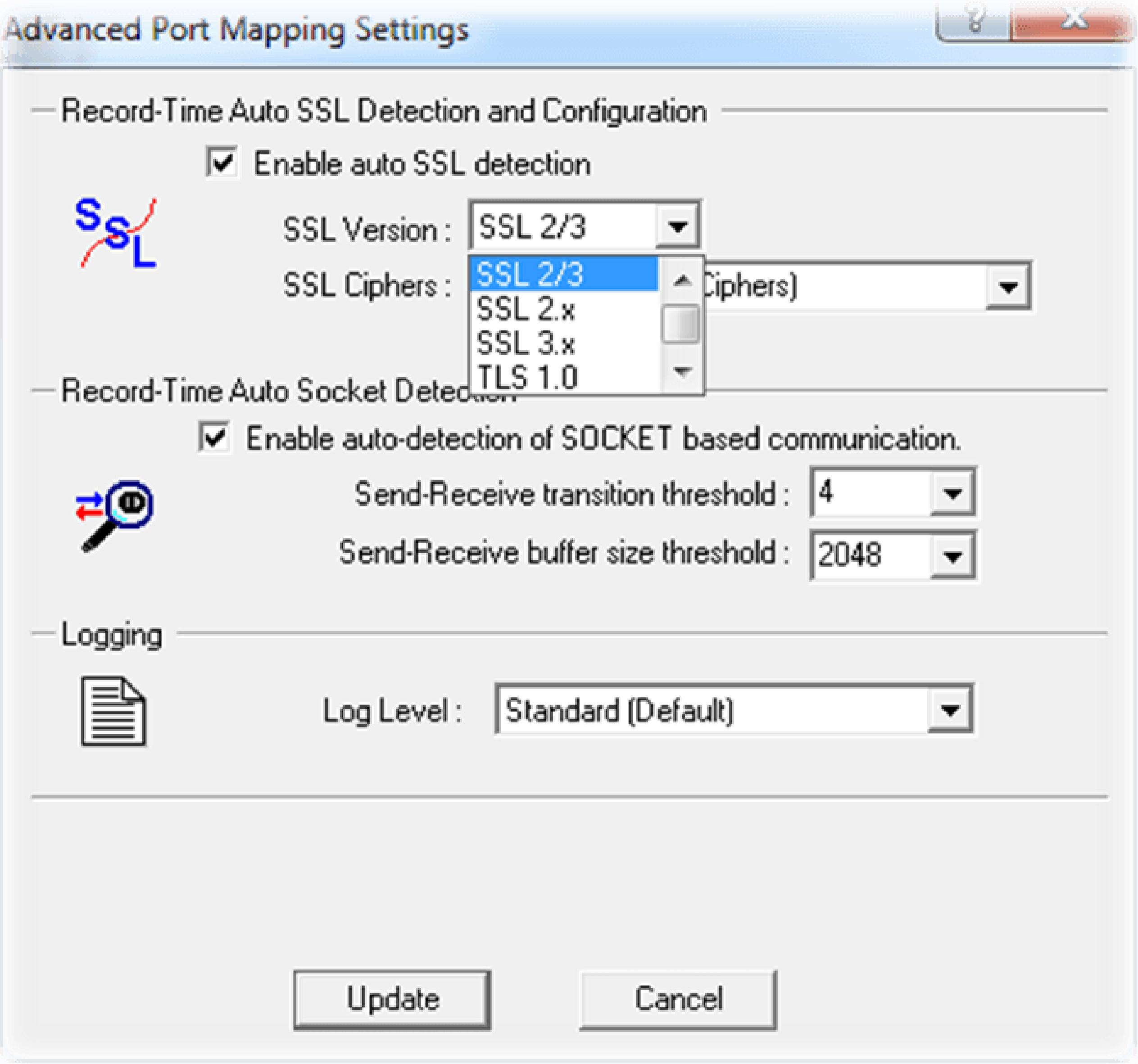
Capture level: Socket level data

Network-level server address mappings for : (All IDs)

Description :

Select a server entry to enable/disable or update.

You can go to Options (as long as you're using Socket level data) and select various options like SSL version or other types of Secure Socket Layer. If you're a beginner level or do not require these options, you can skip. Have a look to get yourself acquainted with the screen.



Advanced Port Mapping Settings

Record-Time Auto SSL Detection and Configuration

☑ Enable auto SSL detection

SSL Version : SSL 2/3

SSL Ciphers : …Ciphers)

Record-Time Auto Socket Detection

☑ Enable auto-detection of SOCKET based communication.

Send-Receive transition threshold : 4

Send-Receive buffer size threshold : 2048

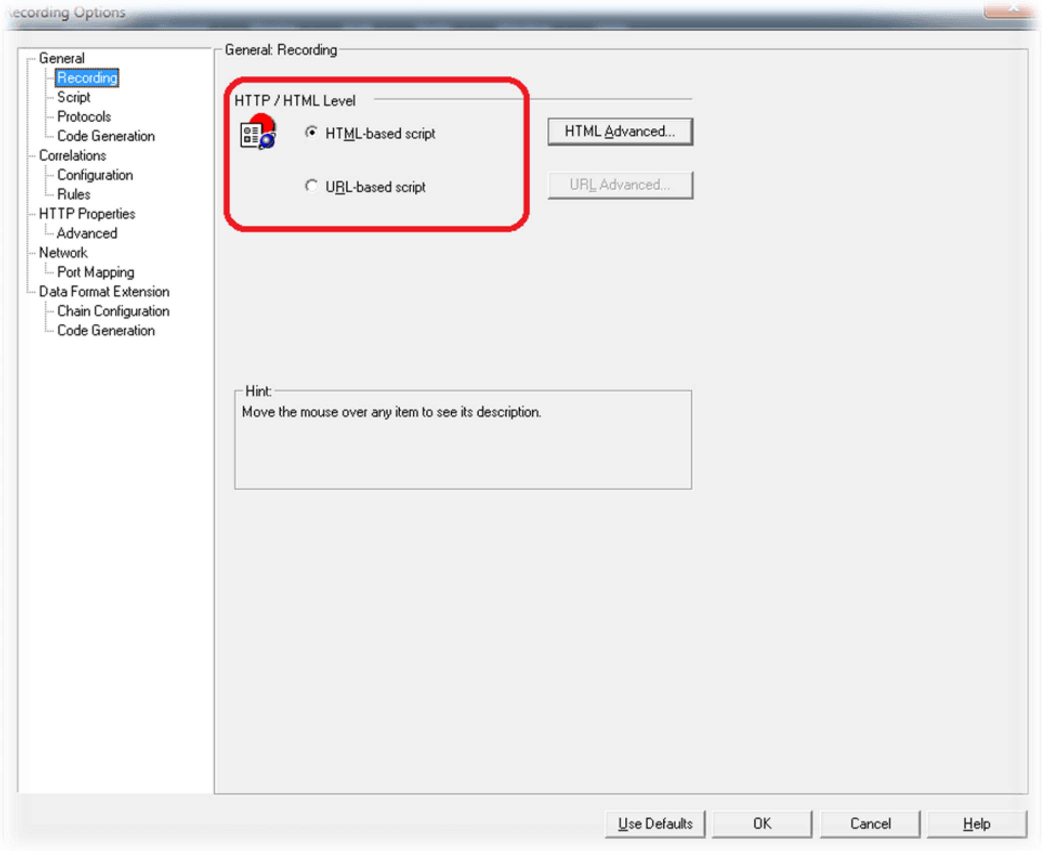
Logging

Log Level : Standard (Default)

Now you're done with most of the Record Time options, let's move to the next topic and understand the difference between HTML and URL based scripting.

Difference between HTML-based and URL-based Scripting

You may have noticed an options to pick either HTML-based script or URL-based script. Have a look at the snapshot for a flashback.



● HTML-based script

○ URL-based script

Hint:
Move the mouse over any item to see its description.

So what is this option and which one to pick?

The HTML-based script is based on user actions, and the scripts contain functions that correspond directly to the action taken. Let's understand example of a small piece of code:

Example:

```
...
web_link("Enterprise Systems Performance",
"Text=Enterprise Systems Performance,"
"Snapshot=t4.inf",
LAST);
```

The URL-based script is based on HTTP requests sent to the server as a result of user actions.

Here is an example of code for URL mode for the same actions performed as above (in HTML mode)

Example:

```
...
web_url("Enterprise Systems Performance",
"URL=http://www.guru99.com/esp.html",
"TargetFrame=",
"Resource=0",
"RecContentType=text/html",
"Referer=http://www.guru99.com/atc?. . . ,
"Snapshot=t4.inf",
"Mode=URL",
LAST);
```

Tip: It's best to experiment yourself before you move forward. Change the record time settings and record same script twice i.e. once with HTML mode and once with URL mode – then compare both. Keep the script short so you can understand the difference.

How do we decide on which mode to use?

Let’s understand the pros and cons of both modes so understand which mode is more suitable under certain situations:

Benefits of HTML Recording

Reduces need to capture dynamic values

- Action tag values and hidden data are NOT hardcoded
- They are retrieved from memory during playback
- If they are dynamic, the VUser still run

Script is only as big as the business process–one step per page

Disadvantages of HTML Recording

Scripts are less scalable

Memory (cache) is searched during playback

- requires more memory
- requires more CPU power

Benefits of URL Recording

Flexibility

- Support for Java Applets and ActiveX objects on the page
- Ability to replay on UNIX

Scalability

- Scripts are more scalable than HTML scripts because they require fewer resources

Disadvantages of URL recording

Scripts require more correlation (nothing is retrieved from the cache)

Context-sensitive checks won’t work (parser is disabled)*

Scripts are large (all images and frames are recorded as separate steps)

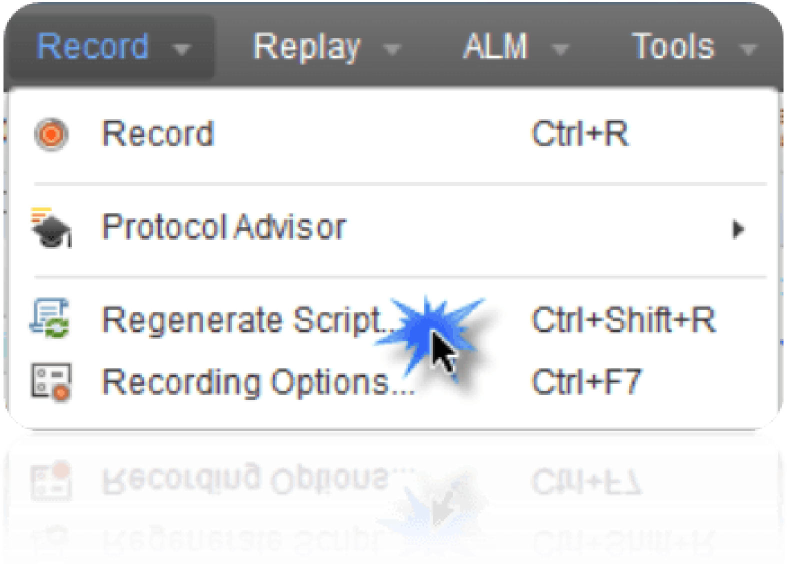
Here is a quick illustration:

HTML Mode	URL Mode
Intuitive and easy to understand	Not as intuitive as the HTML scripts
Scripts are smaller, requests are encapsulated and easy to understand	Scripts are large, containing a call to each image, css, html, etc. thus making it difficult to understand.
Scalable	More scalable and effective for creating a load test

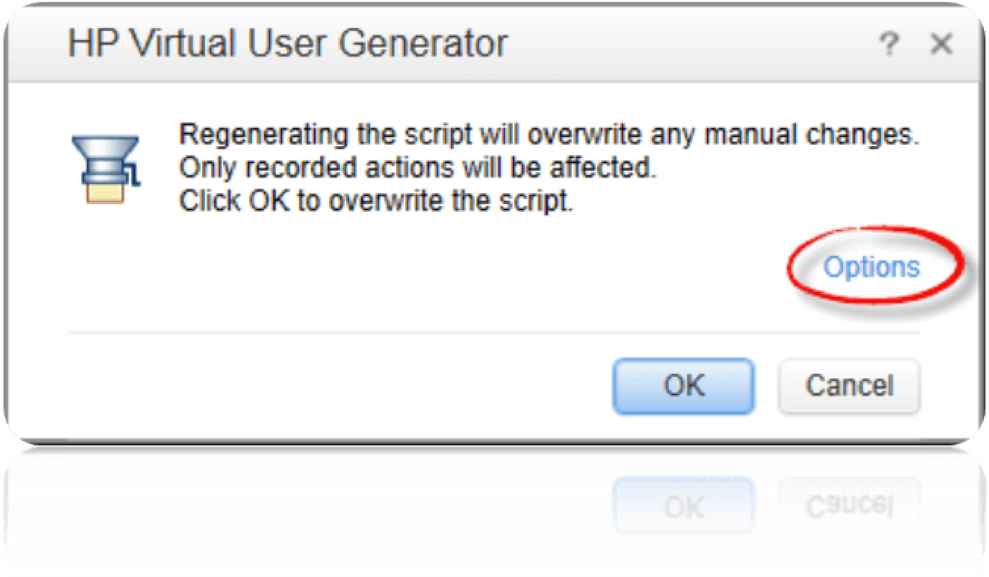
Use of Re-Generate Script

Let’s suppose you want to record the same script you just recorded, but with different record time settings. In such a case, you can use the regenerate script feature.

You can access it under Record => Regenerate Script or with hot key Ctrl+Shift+R


Record
Replay
ALM
Tools
Record Ctrl+R
Protocol Advisor
Regenerate Script... Ctrl+Shift+R
Recording Options... Ctrl+F7

Once you click on the menu, VUGen will give you a warning that your existing script and all changed you've made to your existing script will be lost. The warning message looks like this:

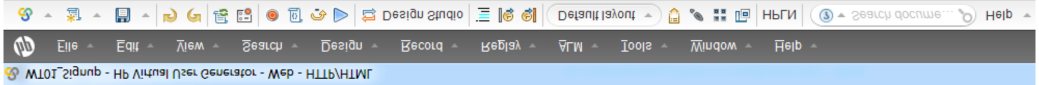


HP Virtual User Generator

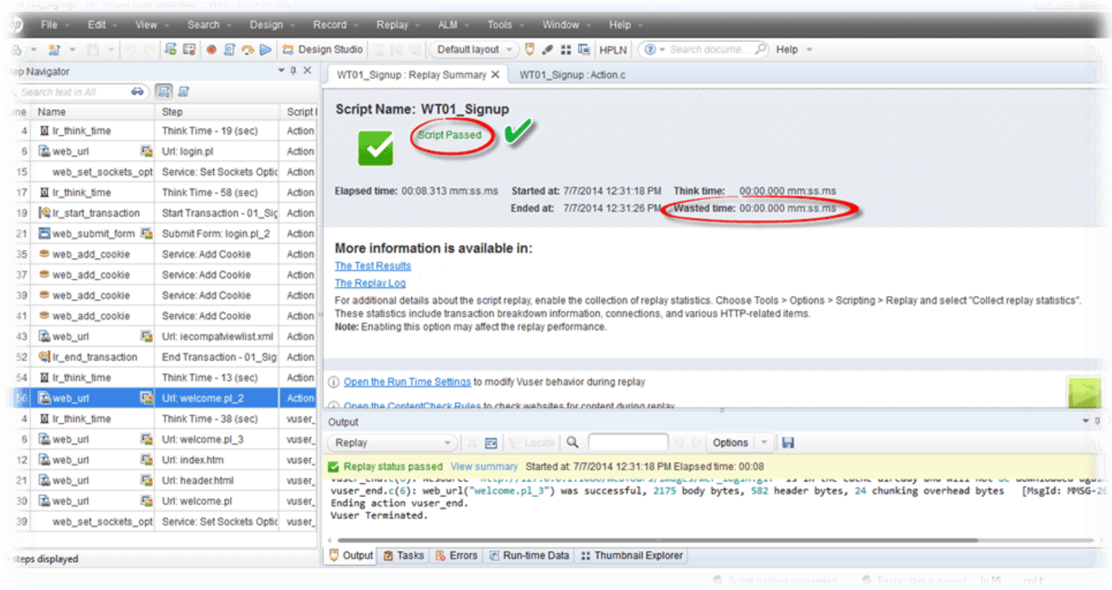
Regenerating the script will overwrite any manual changes.
Only recorded actions will be affected.
Click OK to overwrite the script.

You can also click on Options to open Record Time Options from here.
Click OK to proceed with Re-Generation of script.
Playback a Script and understanding Log

Once you've saved the script, you can click on the ▷ in the toolbar to replay or hit F5.
You can find this button in the toolbar:



You need to ensure the server is running (which is required for application to work properly)
When you replay the script, you'll notice that unlike QuickTest Professional, it doesn't open any browser to replay.
Remember, this execution will simulate only 1 (single) user load on the SUL. The purpose of this execution is to ensure your script is working.



ine	Name	Step	Script
4	lr_think_time	Think Time - 19 (sec)	Action
6	web_url	Url: login.pl	Action
15	web_set_sockets_opt	Service: Set Sockets Optic	Action
17	lr_think_time	Think Time - 58 (sec)	Action
19	lr_start_transaction	Start Transaction - 01_Sic	Action
21	web_submit_form	Submit Form: login.pl_2	Action
35	web_add_cookie	Service: Add Cookie	Action
37	web_add_cookie	Service: Add Cookie	Action
39	web_add_cookie	Service: Add Cookie	Action
41	web_add_cookie	Service: Add Cookie	Action
43	web_url	Url: iecompatviewlist.xml	Action
52	lr_end_transaction	End Transaction - 01_Sig	Action
54	lr_think_time	Think Time - 13 (sec)	Action
56	web_url	Url: welcome.pl_2	Action
4	lr_think_time	Think Time - 38 (sec)	vuser_
6	web_url	Url: welcome.pl_3	vuser_
12	web_url	Url: index.htm	vuser_
21	web_url	Url: header.html	vuser_
30	web_url	Url: welcome.pl	vuser_
39	web_set_sockets_opt	Service: Set Sockets Optic	vuser_

steps displayed

Script Name: WT01_Signup

Script Passed

Elapsed time: 00:08.313 mm:ss.ms **Started at:** 7/7/2014 12:31:18 PM **Think time:** 00:00.000 mm:ss.ms
Ended at: 7/7/2014 12:31:26 PM **Wasted time:** 00:00.000 mm:ss.ms

More information is available in:
The Test Results
The Replay Log
For additional details about the script replay, enable the collection of replay statistics. Choose Tools > Options > Scripting > Replay and select "Collect replay statistics".
These statistics include transaction breakdown information, connections, and various HTTP-related items.
Note: Enabling this option may affect the replay performance.

Open the Run Time Settings to modify Vuser behavior during replay
Open the ContentCheck Rules to check websites for content during replay

Output

Replay Locate Options

Replay status passed View summary Started at: 7/7/2014 12:31:18 PM Elapsed time: 00:08

```
vuser_end.c(6): web_url("welcome.pl_3") was successful, 2175 body bytes, 582 header bytes, 24 chunking overhead bytes   [MsgId: MMSG-2
Ending action vuser_end.
Vuser Terminated.
```

Advance VUGEN: Pamaretization, Transactions, Functions, Run Time Setttings

A recorded script can simulate a virtual user; however, a mere recording may not be enough to replicate the "real user behaviour".

When a script is recorded, it covers single and straight flow of the subject application. Whereas, a real user may perform multiple iterations of any process before he logs out. The delay between clicking buttons (think time) will vary from person to person. Chances are that some real users access your application over DSL and some access it over a dial-up. So, in order to get the real feel of end user, we need to enhance our scripts to be exact match, or at least very close in behaviour to real users.

The above is the most significant consideration when conducting "performance testing", but there is more to a VU Script. How will you gauge the precise amount of time taken by a VUser when SUL is undergoing a performance test? How would you know if the VUser has passed through or failed at certain point? What is the cause behind the failure, whether some backend process failed or the server resources were limited?

We need to enhance our script to help answer all above questions.

- Using Transactions
- Understanding Think Time, Rendezvous Points and Comments
- Inserting Functions through menu
- Parameterization and it's configuration
- Run Time Settings and their impact on VU simulation
    - Run Logic
    - Pacing
    - Log
- Think Times
- Speed Simulation
- Browser Emulation
- Proxy

Using Transactions

Transactions are mechanics to measure server response time for any operation. In simple words, the use of "Transaction" helps measure the time taken by the system for a particular request. It can be as small as a click of a button or an AJAX call upon losing focus from the text box.

Applying transactions is straightforward. Just write one line of code before request is made to the server and close the transaction when the request ends. LoadRunner requires only a string as transaction name.

To open a transaction, use this line of code:

lr_start_transaction("Transaction Name");

To close the transaction, use this line of code:

lr_end_transaction("Transaction Name", <status>);

The <status> tells LoadRunner whether this particular transaction was successful or unsuccessful. The possible parameters could be:

- LR_AUTO
- LR_PASS
- LR_FAIL

Example:

lr_end_transaction("My_Login", LR_AUTO);
lr_end_transaction("001_Opening_Dashboard Name", LR_PASS);
lr_end_transaction("Business_Workflow_Transaction Name", LR_FAIL);

Points to note:

- Don't forget, you are working with "C" and that is a case-sensitive language.
- Period (.) character is not allowed in transaction name, although you can use spaces and underscore.
- If you've branched your code well and added checkpoints to verify the response from the server, you can use custom error handling, such as, LR_PASS or LR_FAIL. Otherwise, you can use LR_AUTO and LoadRunner will automatically handle server error (HTTP 500, 400 etc.)
- When applying transactions, ensure there is no **think_time** statement being sandwiched or otherwise your transaction will always include that period.

- Since LoadRunner requires a constant string as transaction name, a common problem when applying transaction is mismatch of string. If you give a different name when opening and closing a transaction, you will at least 2 errors. Since the transaction you opened was never closed, LoadRunner will yield an error. Besides, the transaction you are trying to close was never opened, hence resulting an error.
- Can you use your intelligence and answer to yourself which of the above error will be reported first? To validate your answer, why not make your own mistake? If you had answered right, you are on track. If you answered wrong, you need to focus.
- Since LoadRunner automatically takes care of synchronization of requests and response, you will not have to worry about response when applying transactions.

Understanding Think Time, Rendezvous Points and Comments

Rendezvous Points

Rendezvous Points means "meeting points". It is just one line of statement that tells LoadRunner to introduce concurrency. You insert rendezvous points into VUser scripts to emulate heavy user load on the server.

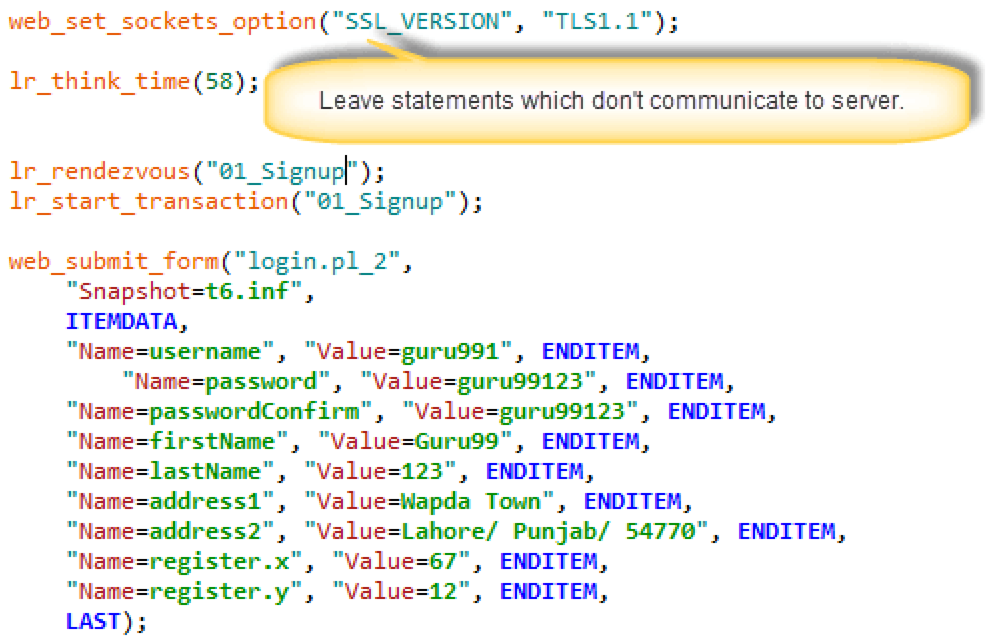
Rendezvous points instruct VUser to wait during test execution for multiple VUser to arrive at a certain point, so that they may concurrently perform a task. For example, to emulate peak load on the bank server, you can insert a rendezvous point instructing 100 VUser to deposit cash into their accounts at the same time. This can be achieved easily using rendezvous.

If the rendezvous points are not places correctly, the VUser will be accessing different parts of the application – even for the same script. This is because every VUser gets different response time and hence few users lag behind.

Syntax: lr_rendesvous("Logical Name");

Best Practices:

- Prefix a rendezvous point with "rdv_" for better code readability; e.g. "rdv_Login"
- Remove any immediate think time statements
- Applying rendezvous points in a script view (after recording)



```
web_set_sockets_option("SSL_VERSION", "TLS1.1");

lr_think_time(58);

lr_rendezvous("01_Signup");
lr_start_transaction("01_Signup");

web_submit_form("login.pl_2",
    "Snapshot=t6.inf",
    ITEMDATA,
    "Name=username", "Value=guru991", ENDITEM,
        "Name=password", "Value=guru99123", ENDITEM,
    "Name=passwordConfirm", "Value=guru99123", ENDITEM,
    "Name=firstName", "Value=Guru99", ENDITEM,
    "Name=lastName", "Value=123", ENDITEM,
    "Name=address1", "Value=Wapda Town", ENDITEM,
    "Name=address2", "Value=Lahore/ Punjab/ 54770", ENDITEM,
    "Name=register.x", "Value=67", ENDITEM,
    "Name=register.y", "Value=12", ENDITEM,
    LAST);
```

Leave statements which don't communicate to server.

Comments

Add comments to describe an activity, a piece of code or a line of code. Comments help make the code understandable for anyone referring to it in the future. They provide information about specific operation and separate two sections for distinction.

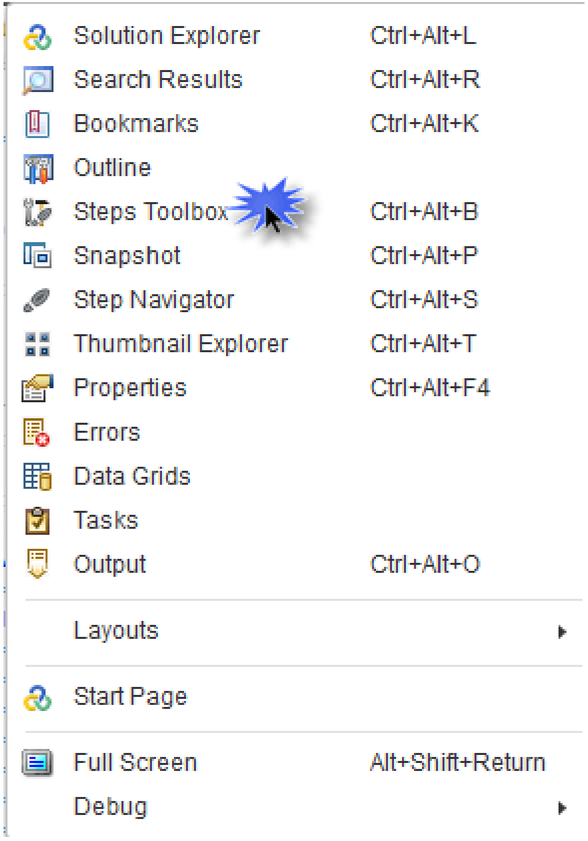
You can add comments

- While recording (using tool)
- After recording (directly writing in code)

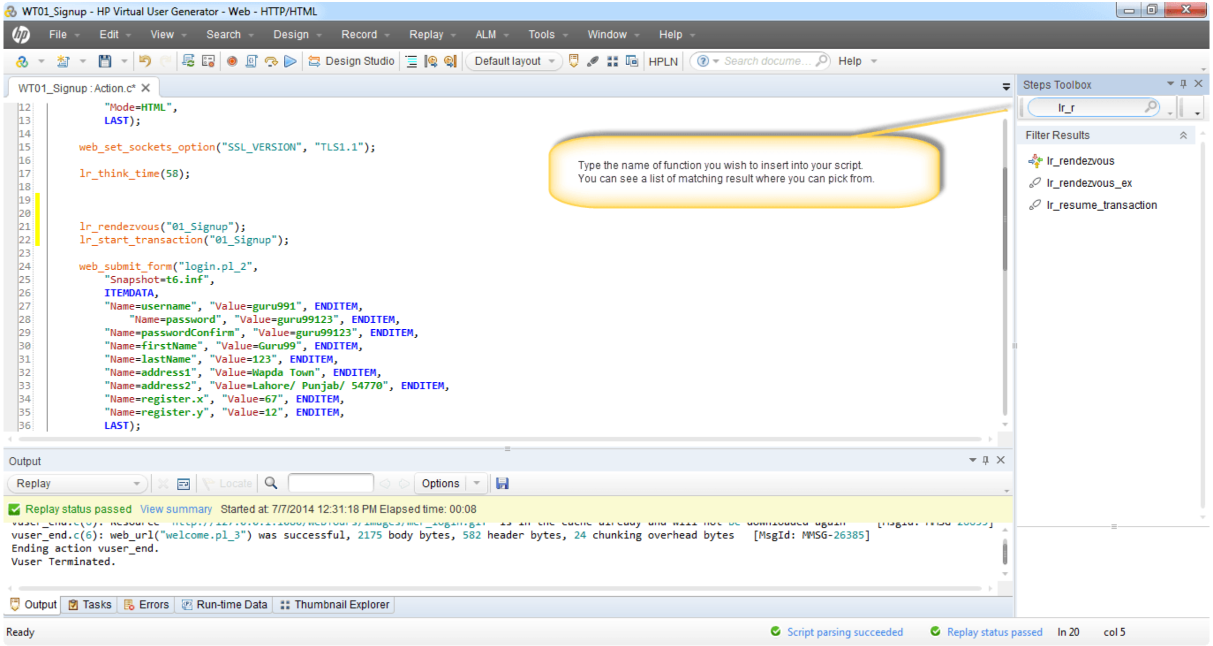
Best Practice: Mark any comments on the top of each script file

Inserting Functions through menu

While you can directly write simple lines of code, you may need a clue to recall a function. You can also use Steps Toolbox (known as Insert Function prior to version 12) to find and insert any function directly into your script. You can find Steps Toolbar under View àSteps Toolbox.



This will open a side window, look at the snapshot:



```
        "Mode=HTML",
        LAST);

    web_set_sockets_option("SSL_VERSION", "TLS1.1");

    lr_think_time(58);



    lr_rendezvous("01_Signup");
    lr_start_transaction("01_Signup");

    web_submit_form("login.pl_2",
        "Snapshot=t6.inf",
        ITEMDATA,
        "Name=username", "Value=guru991", ENDITEM,
            "Name=password", "Value=guru99123", ENDITEM,
        "Name=passwordConfirm", "Value=guru99123", ENDITEM,
        "Name=firstName", "Value=Guru99", ENDITEM,
        "Name=lastName", "Value=123", ENDITEM,
        "Name=address1", "Value=Wapda Town", ENDITEM,
        "Name=address2", "Value=Lahore/ Punjab/ 54770", ENDITEM,
        "Name=register.x", "Value=67", ENDITEM,
        "Name=register.y", "Value=12", ENDITEM,
        LAST);
```

Type the name of function you wish to insert into your script. You can see a list of matching result where you can pick from.

Replay status passed View summary Started at: 7/7/2014 12:31:18 PM Elapsed time: 00:08

```
vuser_end.c(6): web_url("welcome.pl_3") was successful, 2175 body bytes, 582 header bytes, 24 chunking overhead bytes   [MsgId: MMSG-26385]
Ending action vuser_end.
Vuser Terminated.
```

Parameterization and it's configuration

A **parameter** in VUGen is a container that contains a recorded value that is replaced for various users.

During the execution of the script (in VUGen or Controller), the value from an external source (like .txt, XML or database) substitutes the previous value of the parameter.

Parameterization is useful in sending dynamic (or unique) values to the server, for example; a business process is desired to run 10 iterations but picking unique user name every time.

It also helps in stimulating real-like behavior to the subject system. Have a look at below example:

Problem examples:

Business process works only for the current date which comes from the server, hence can't be passed as a hardcoded request.

Sometimes, the client application passes a Unique ID to the server (for example session_id) for the process to continue (even for a single user) – In such a case, parameterization helps.

Often, the client application maintains a cache of data being sent to and from the server. As a result, server is not receiving a real user behavior (in case server runs different algorithm depending upon search criteria). While VUser script will execute successfully, the performance statistics drawn will not be meaningful. Using different data through parameterization helps emulates server side activity (procedures etc.) and exercises the system.

A date that is hard-coded in the VUser during recording may no longer be valid when that date has passed. Parameterizing the date allows VUser execution to succeed by replacing the hard-coded date. Such fields or requests are the right candidates for parameterization.

Please be patient. The Video will load in some time. If you still face issue viewing video click here

Run Time Settings and their impact on VU simulation

Run Time Settings bear as much significant as your VUGen Script. With varying configurations, you can obtain different test designs. This is why, you may end up in non-repeatable results if Run Time Settings are not consistent. Let's discuss each attribute one by one.

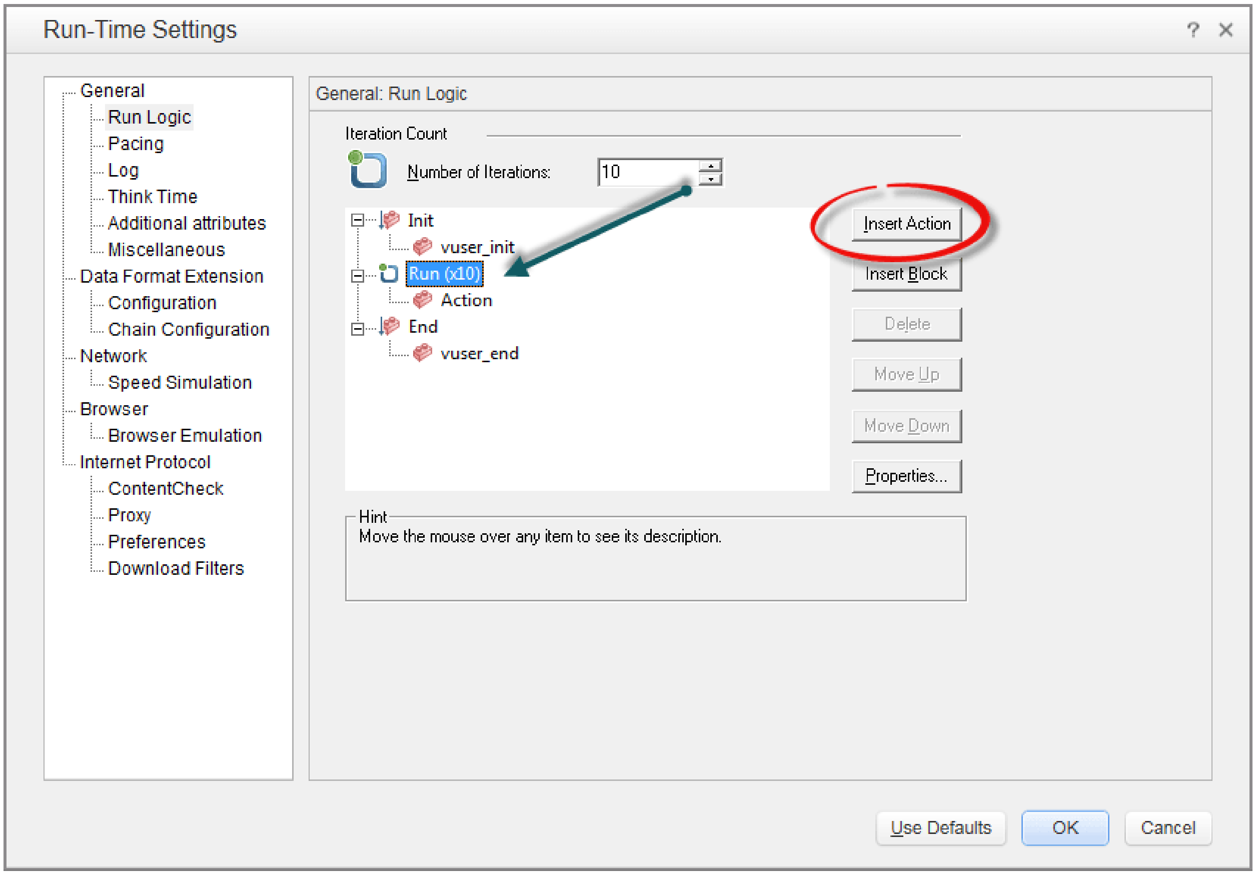
Run Logic

Run Logic defines the number of times all actions will be executed, except vuser_init and vuser_end.

Probably this makes clearer why LoadRunner suggests keeping all the Login code within vuser_init, and Logout part in vuser_end, both exclusively.

If you've created multiple actions, let's say, Sign in, Open Screen, Calculate Rental, Submit Funds, Check Balance and logout, then, below scenario will take place for each VUser:

All VUsers will login, execute Open Screen, Calculate Rental, Submit Funds, Check Balance – then – again Open screen, Calculate rentals…and so on – iterating 10 times – followed by logout (once).



Number of Iterations: 10

Hint
Move the mouse over any item to see its description.

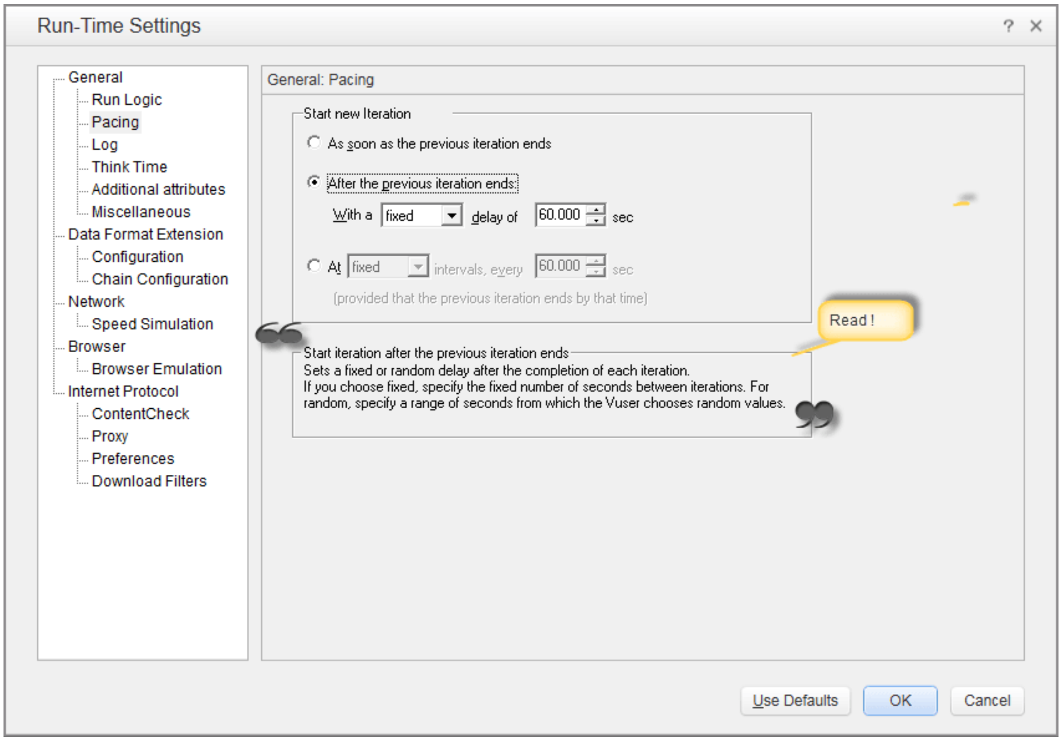
This is a powerful setting enabling to act more like a real user. Remember, a real user does not login and logouts every time – he, usually, repeats same steps.

How many times do you click “inbox” when checking you email before logout?

Pacing

This is important. Mostly people are unable to understand the different between pacing and think time. The only difference is, **“pacing refers to the delay between iterations” whereas think time is the delay between any 2 steps.**

Recommended setting depends upon the test design. However, if you are looking to have aggressive load, consider opting “As soon as the previous iteration ends”



General: Pacing

Start new Iteration

○ As soon as the previous iteration ends

● After the previous iteration ends:

With a fixed delay of 60.000 sec

○ At fixed intervals, every 60.000 sec

(provided that the previous iteration ends by that time)

Start iteration after the previous iteration ends

Sets a fixed or random delay after the completion of each iteration.
If you choose fixed, specify the fixed number of seconds between iterations. For random, specify a range of seconds from which the Vuser chooses random values.

Log

A log (as generally understood) is a bookkeeping of all events while you run LoadRunner. You can enable log to know what’s happening between your application and your server.

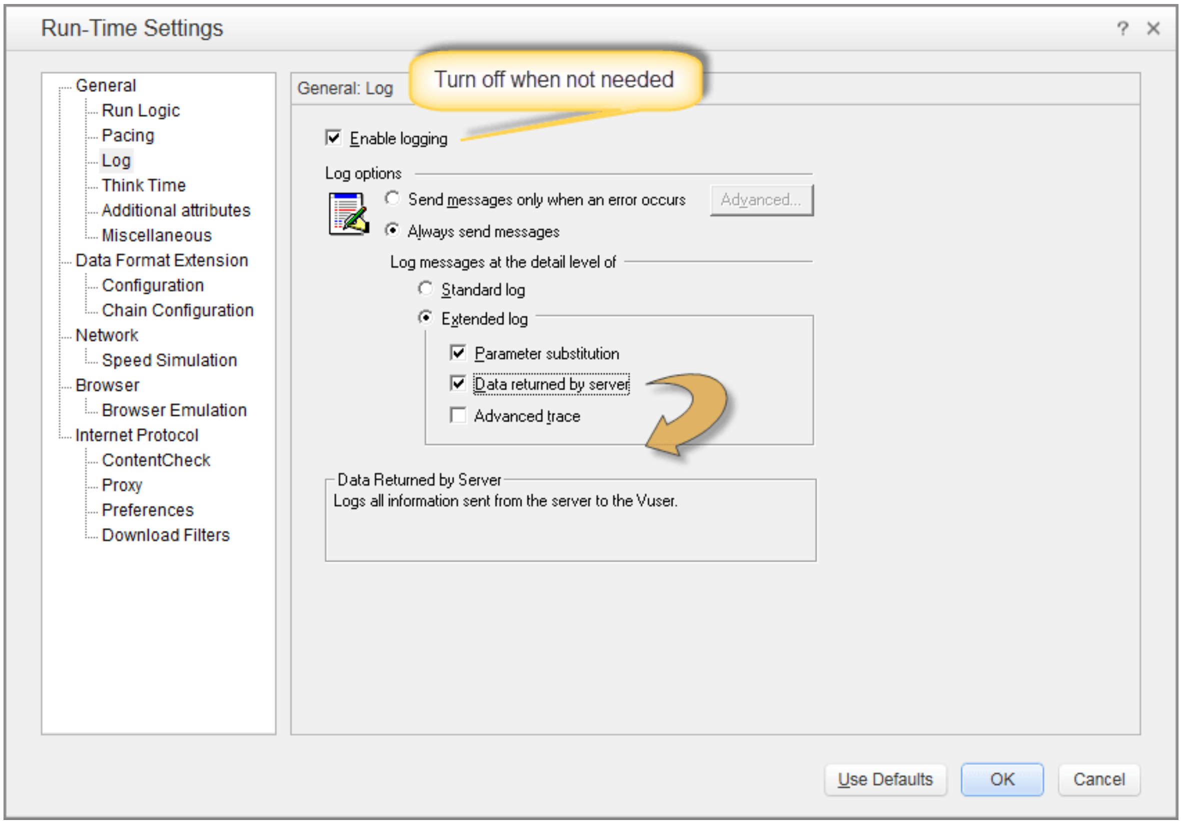
LoadRunner gives powerful logging mechanism which is robust and scalable on its own. It allows you to keep only “Standard Log” or a detailed, configurable extended log or disable it altogether.

A standard log is informative and easily understandable. It contains just the right amount of knowledge you will generally require troubleshooting your VUser scripts.

In the case of Extended Log, all the Standard log information is a subset. Additionally, you can have parameter substitution. This tells LoadRunner component to include complete information of all the parameters (from parameterization) including requests, as well as response data.

If you include “Data Returned by Server” then your log will go in length. This will include all the HTML, tags, resources, non-resources information included right within the log. The option is good only if you need serious troubleshooting. Usually, this makes the log file very big in size and not easily comprehendible.

As you could have guess by now if you opt for “Advance Trace”, your log file will be massive. You must give it a try. You will notice the amount of time taken by VUGen has also increased significantly, although this will have no impact on the transaction response time reported by VUGen. However, this is very advance information and maybe useful if you understand the subject application, the client to server communication between your application and hardware as well as protocol level details. Usually, this information is dead by essence since it requires extreme efforts to understand and troubleshoot.



General: Log

☑ Enable logging

Log options

○ Send messages only when an error occurs

● Always send messages

Log messages at the detail level of

○ Standard log

● Extended log

☑ Parameter substitution

☑ Data returned by server

□ Advanced trace

Data Returned by Server

Logs all information sent from the server to the Vuser.

Tips:

- No matter how much time VUGen takes when log is enabled, it has no impact on the transaction response time. HP calls this phenomenon as “state of the art technology.”
- Disable log if it is not required.
- Disable log when you are finished with your scripts. Including scripts with logging enabled will cause controller to run slower and report nagging messages.
- Disabling log will increase the capacity of the maximum number of users you can simulate from LoadRunner.
- Consider using “Send message only when error occurs” – this will mute unnecessary information messages and report only error related messages.

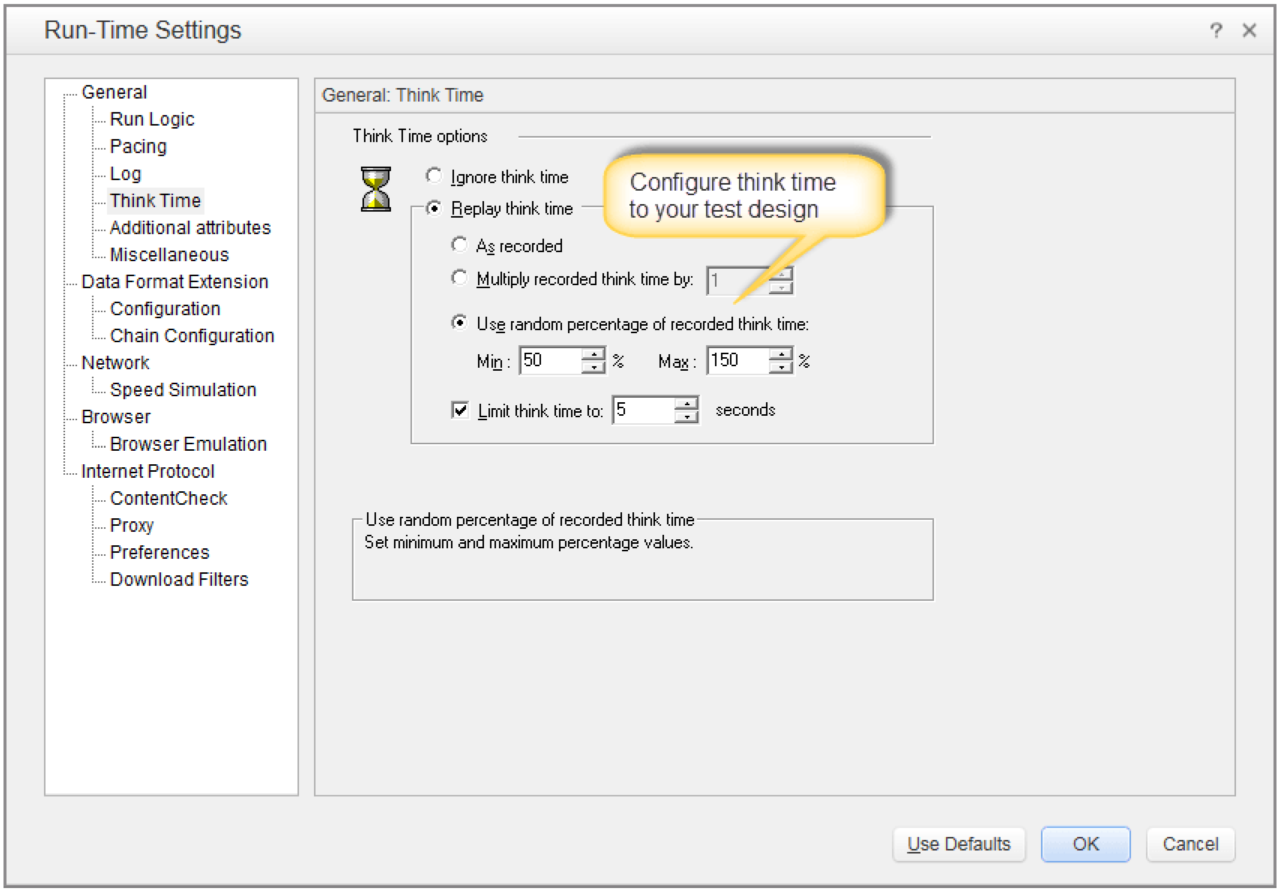
Think Times

Think Time is simply the delay between two steps.

Think Time helps replicates user behavior since no real user can use any application like a machine (VUGen).

VUGen generates think time automatically. You still have complete control to remove, multiply or fluctuate the duration of think time.

To understand more, for example, a user may open a screen (that is a response followed by a request) and then provide it is username and password before hitting enter. The next interaction of the application to the server will happen when he clicks “Sign In”. The time a user took to type his username and password is Think Time in LoadRunner.



General: Think Time

Think Time options

○ Ignore think time

● Replay think time

○ As recorded

○ Multiply recorded think time by: 1

● Use random percentage of recorded think time:

Min : 50 % Max : 150 %

☑ Limit think time to: 5 seconds

Use random percentage of recorded think time

Set minimum and maximum percentage values.

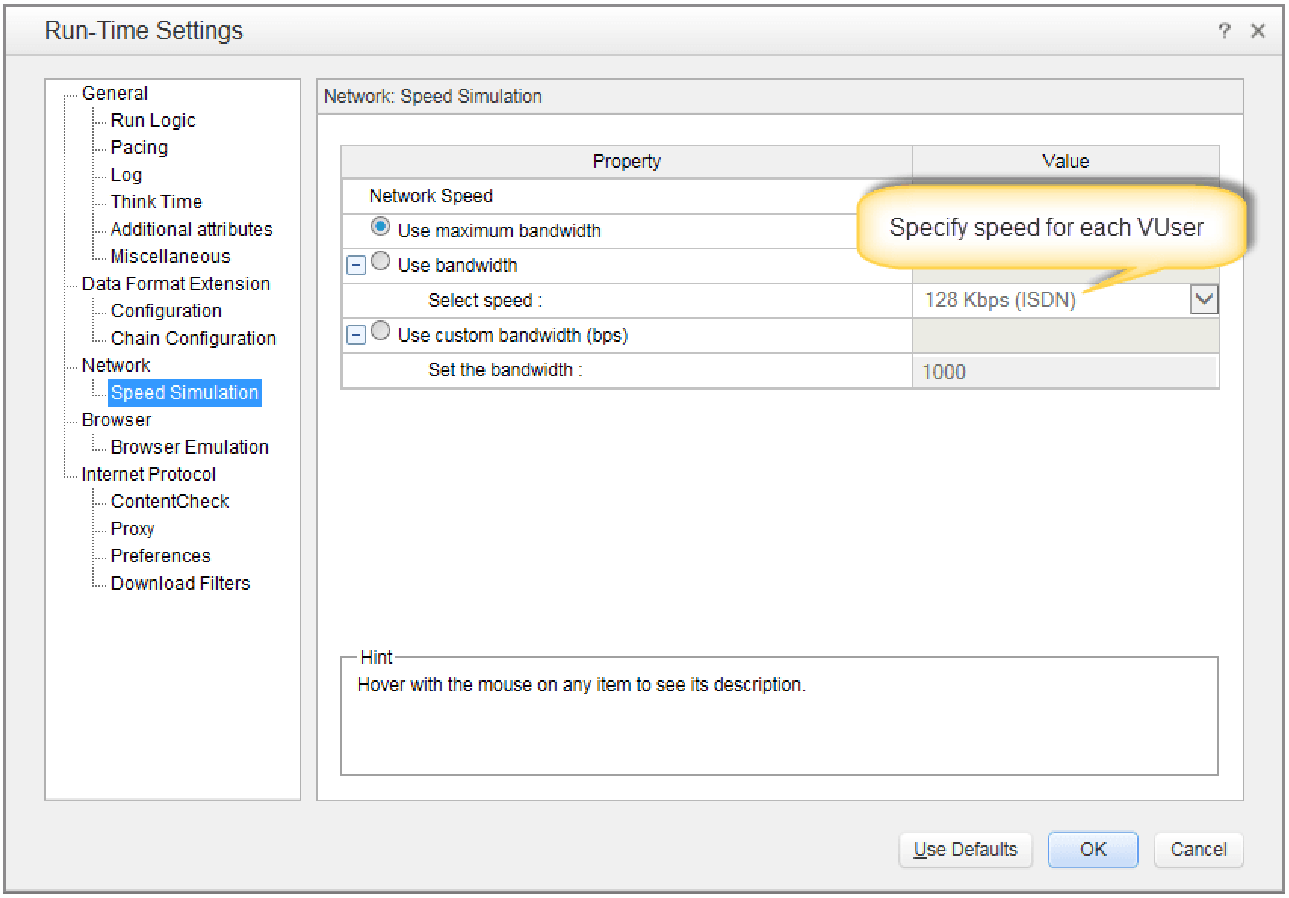
If you are looking to simulate aggressive load on the application, consider disabling think time completely. However, to simulate a real like behavior, you can “User Random Think Time” and set the percentages as desired. Consider using Limit Think Time to a legitimate period. Usually, 30 seconds is fairly good enough.

Speed Simulation

Speed simulation simply refers to bandwidth capacity for each client machine.

Since we are simulating thousands of VUser’s through LoadRunner, it is amazing how simple LoadRunner has made to control the bandwidth/network speed simulation.

If you are customers access your application over 128 Kbps, you can control it from here. You will get to simulate “real like behavior” which should help getting the right performance statistics.



Network: Speed Simulation

Property	Value
Network Speed	
● Use maximum bandwidth	
○ Use bandwidth	
Select speed :	128 Kbps (ISDN)
○ Use custom bandwidth (bps)	
Set the bandwidth :	1000

Specify speed for each VUser

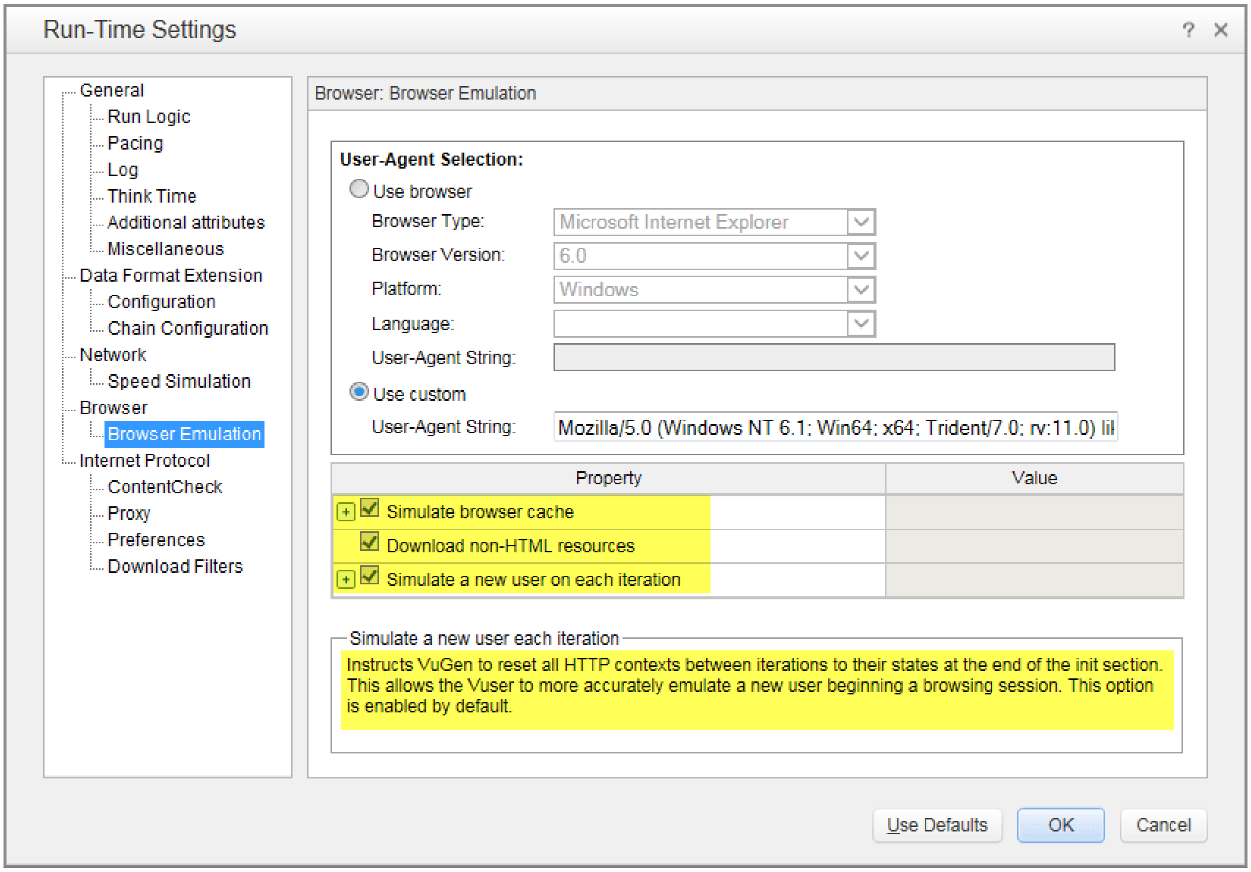
Hint

Hover with the mouse on any item to see its description.

The best recommendation is to set to Use maximum bandwidth. This will help disregard any network related performance bottlenecks and focus on any potential issues in the application first. You can always run the test multiple times to see varying behavior under different circumstances.

Browser Emulation

User experience does not depend upon the browser an end user is using. Clearly, this is beyond the scope of Performance measures. However, you can choose which browser you wish to emulate.



User-Agent Selection:

○ Use browser

Browser Type: Microsoft Internet Explorer

Browser Version: 6.0

Platform: Windows

Language:

User-Agent String:

● Use custom

User-Agent String: Mozilla/5.0 (Windows NT 6.1; Win64; x64; Trident/7.0; rv:11.0) lil

Property	Value
+ ☑ Simulate browser cache	
☑ Download non-HTML resources	
+ ☑ Simulate a new user on each iteration	

Simulate a new user each iteration

Instructs VuGen to reset all HTTP contexts between iterations to their states at the end of the init section. This allows the Vuser to more accurately emulate a new user beginning a browsing session. This option is enabled by default.

Can you answer to yourself when exactly it will really matter for you to select the right browser in this configuration?

You will use this configuration if you are subject application is a web application, returning different responses for different browsers. For example, you get to see different images and contents for IE and Firefox etc.

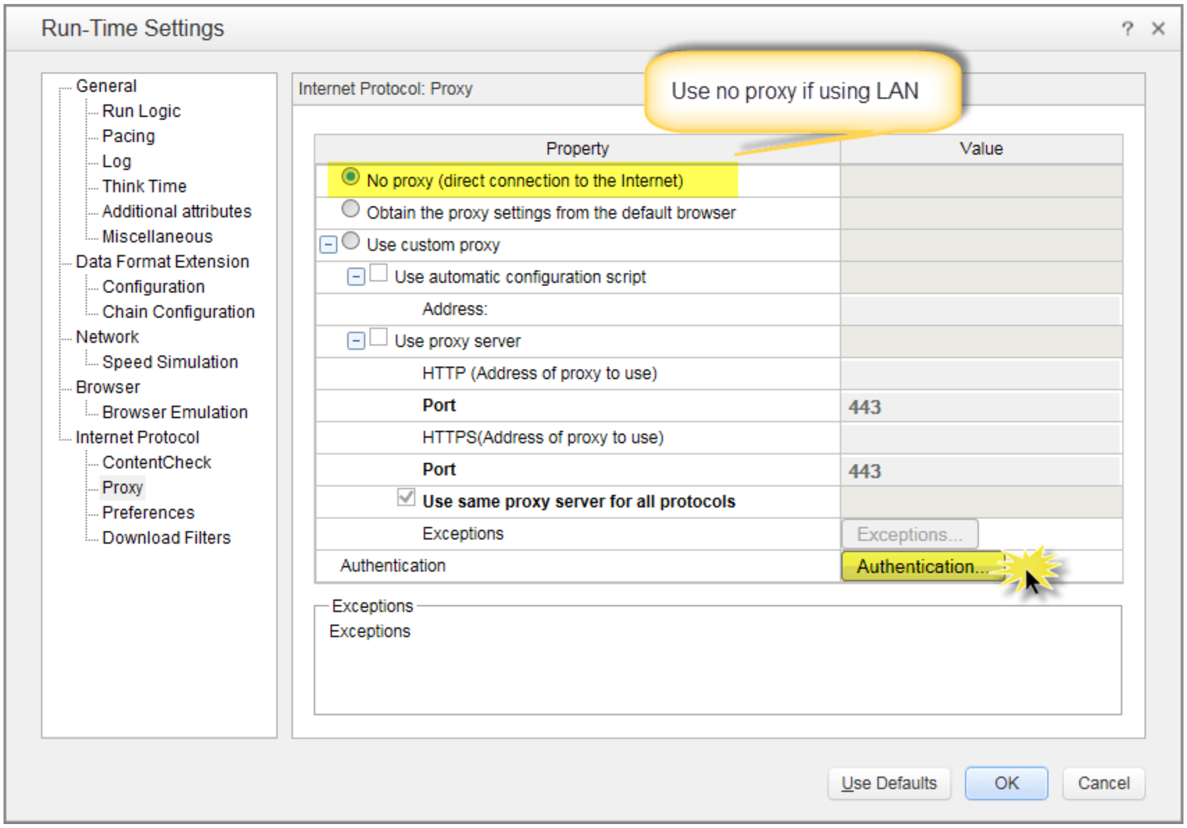
Another important setting is Simulate browser cache. If you want to gauge the response time when cache enabled, check this box. If you are looking for worst case situation, this is obviously not a consideration.

Download non-HTML resources will let LoadRunner download any CSS, JS and other rich media. This should be remained checked. However, if you which to eliminate this from your performance test design, you can uncheck this.

Proxy

It is best to eliminate proxy completely from your test environment – this will make the test results unreliable. However, you might face situations where it is inevitable. In such a situation, LoadRunner does facilitate you with proxy settings.

You will be working (or should be working) with No proxy setting. You can obtain it from your default browser. However, don't forget to check which browser is set to default and what proxy configuration for default browser is.



Use no proxy if using LAN

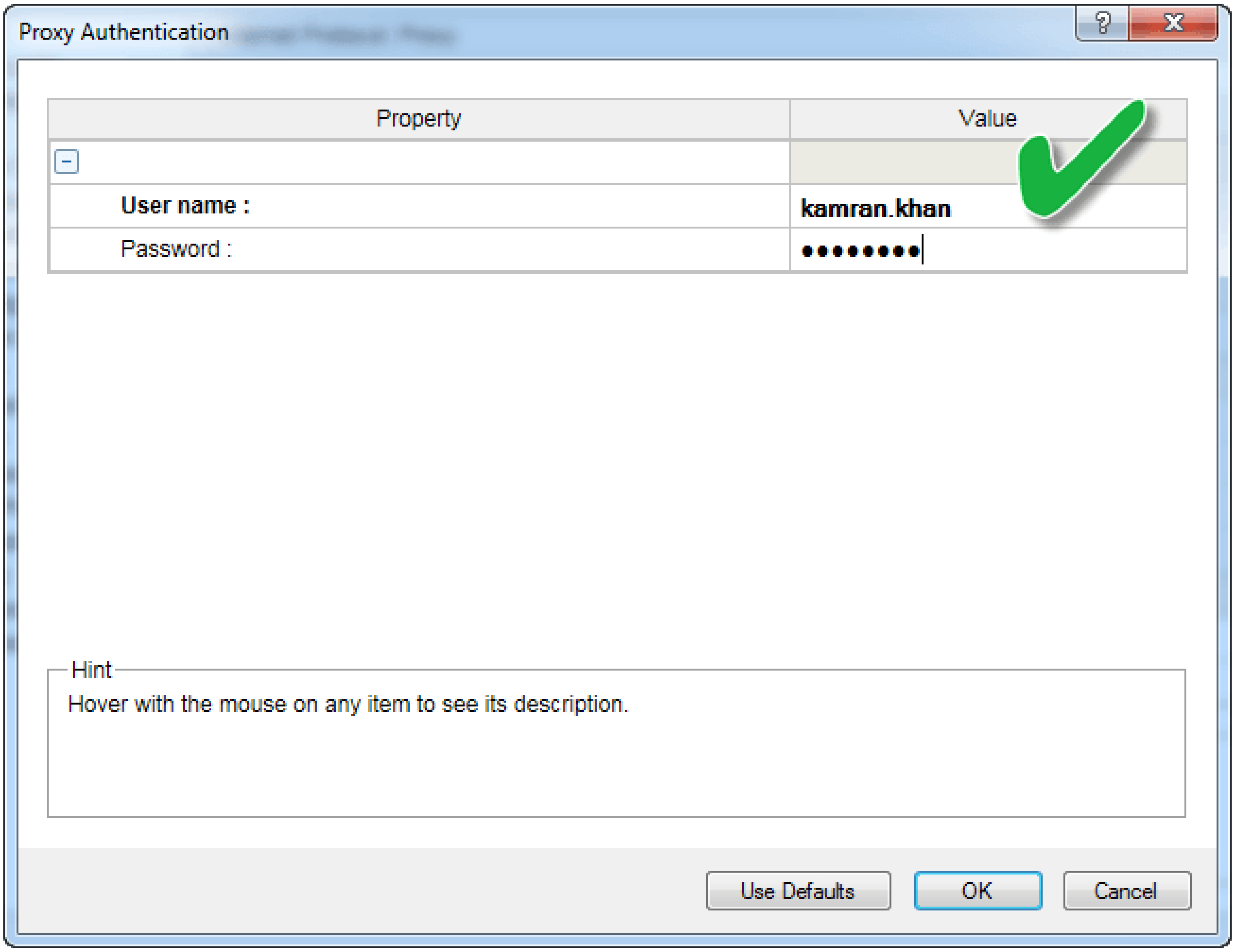
Internet Protocol: Proxy

Property	Value
● No proxy (direct connection to the Internet)	
○ Obtain the proxy settings from the default browser	
⊟ ○ Use custom proxy	
⊟ □ Use automatic configuration script	
Address:	
⊟ □ Use proxy server	
HTTP (Address of proxy to use)	
Port	443
HTTPS(Address of proxy to use)	
Port	443
☑ **Use same proxy server for all protocols**	
Exceptions	Exceptions...
Authentication	Authentication...

Exceptions

Exceptions

If you are using a proxy and it requires authentication (or a script) then you can click on the Authenticate button which leads to a new window. Refer to below screenshot.



Proxy Authentication [redacted]

Property	Value
⊟	
User name :	**kamran.khan**
Password :	••••••••

Hint

Hover with the mouse on any item to see its description.