Controller

Controller – This is the main part of LoadRunner. Both controller and analysis are licensed.

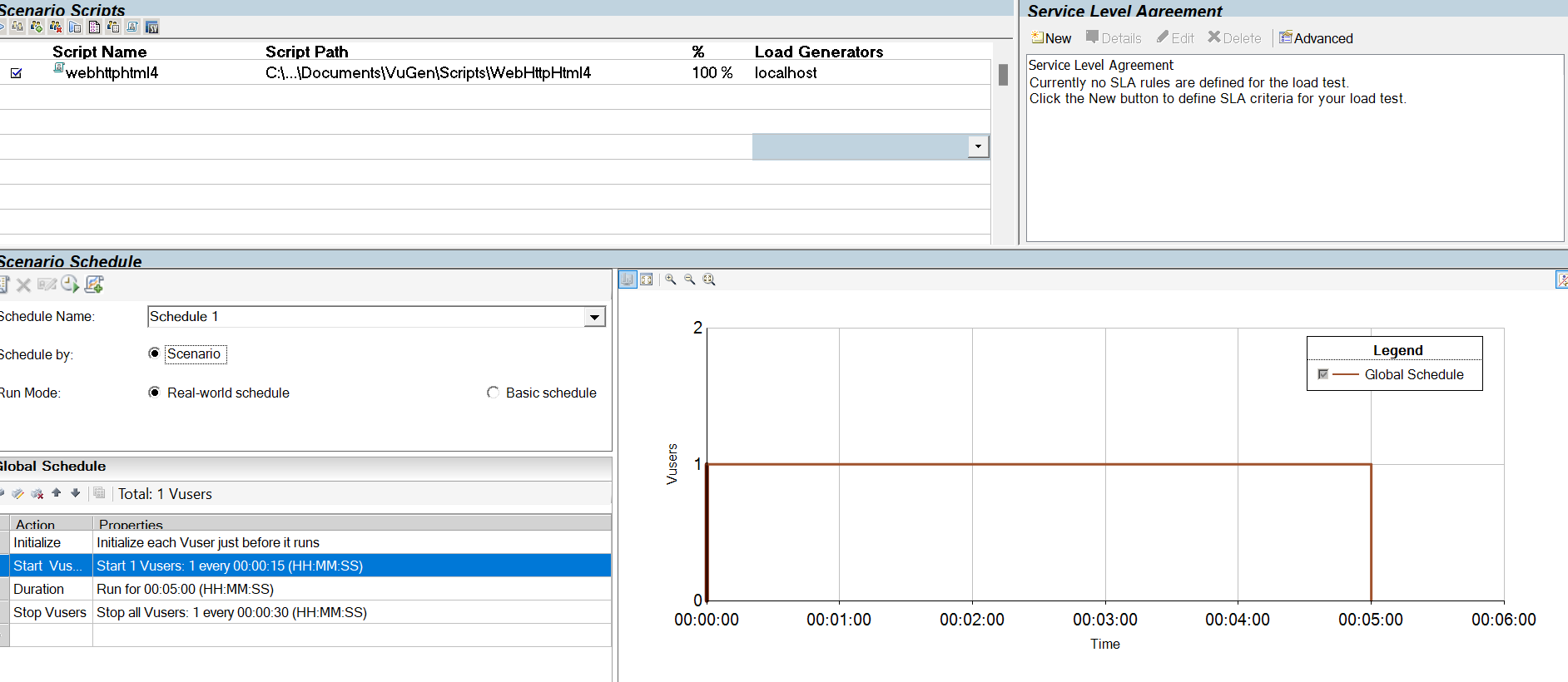
First Step would be to select a scenario.

Scenarios are divided into

* **Manual Scenario** - Here user can control the load applied, decide when user starts/stops, duration of execution
* **Goal-Oriented Scenario** – Define the goal that needs to be achieved and then load runner will design the scenario for the user.

## Designing Scripts

* Here we provide the details of the scripts that should be executed.
* We can add more scripts by selecting the path of the script.
* User can also mention how the load distribution should happen for each script.



**Scenario schedule**

* Real world schedule – Specific duration
* Basic Schedule – Picks the iteration based on what was defined during script creation.

# Types of Scenarios

‘Scenario Preparation’ is one of the important steps in LoadRunner and it comes after the completion of test scripts. When the test scripts work properly on the local machine then you can upload them in Controller

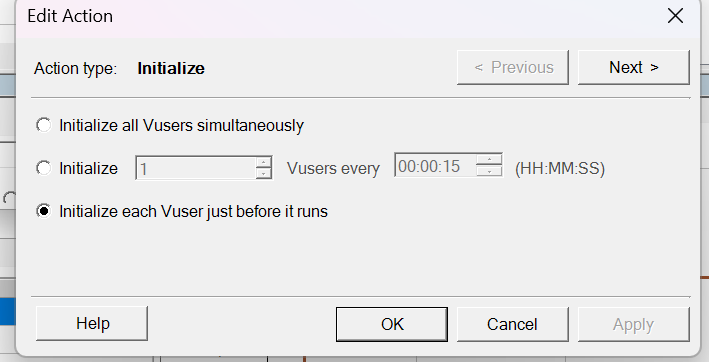
1. Manual Scenario
2. Goal Oriented Scenarios

## Manual Scenario

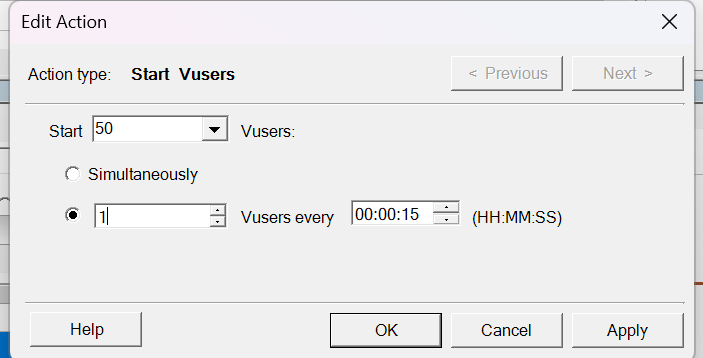
In a manual scenario, you need to design the scenario as per the given performance metrics manually. While designing the scenario, you have to provide inputs like how many users to run, how long, and how the Vusers have to be started and ended. Based on the scenario, LoadRunner executes the test.

Things to provide for manual scenario

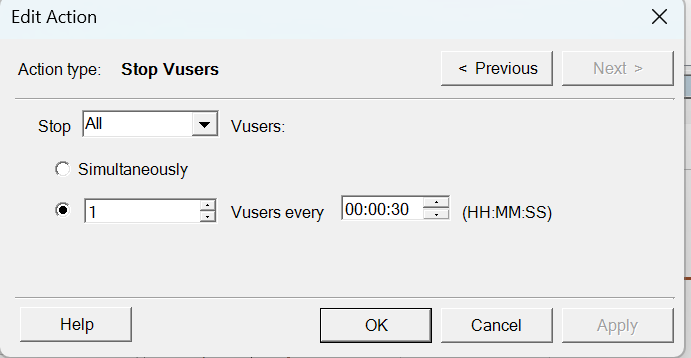
1. Global Schedule
   1. Setting data for initializer – This applies to startInit in Vugen



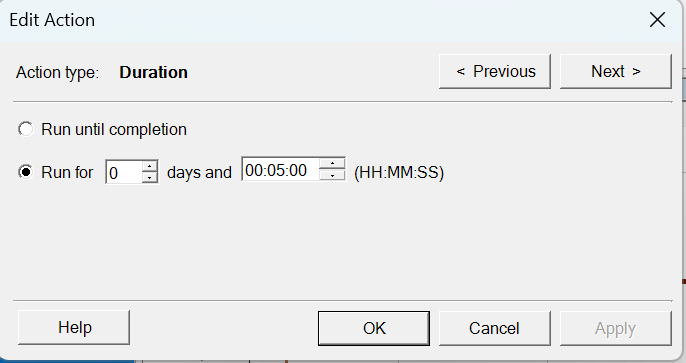
* 1. Start vuser – This is for Actions



* 1. Stop user – This is for vuser\_end

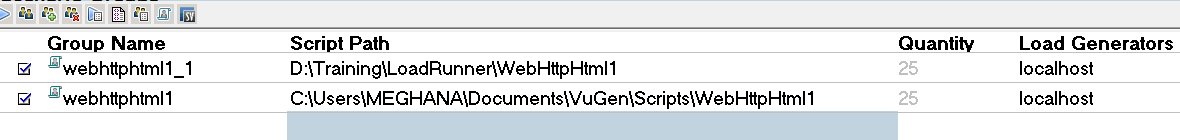


* 1. Duration – How long the test should be executed.



1. Selecting scripts

Multiple scripts can be selected and the load gets distributed based on the value selected initially



1. Load Generator

The VuGen can play any script back, but it can only do so one at a time and therefore cannot generate significant loads. This task is performed by the Load Generator. The Load Generator creates multiple Vusers (each of which is implemented either as a thread or a process, depending on settings) and assigns a script to each. Each Vuser then runs its script for the duration of the test. The Load Generator is not an interactive program; it runs in the background with no user interface. It starts and stops the Vusers, and applies any options and settings on the instructions of the controller.

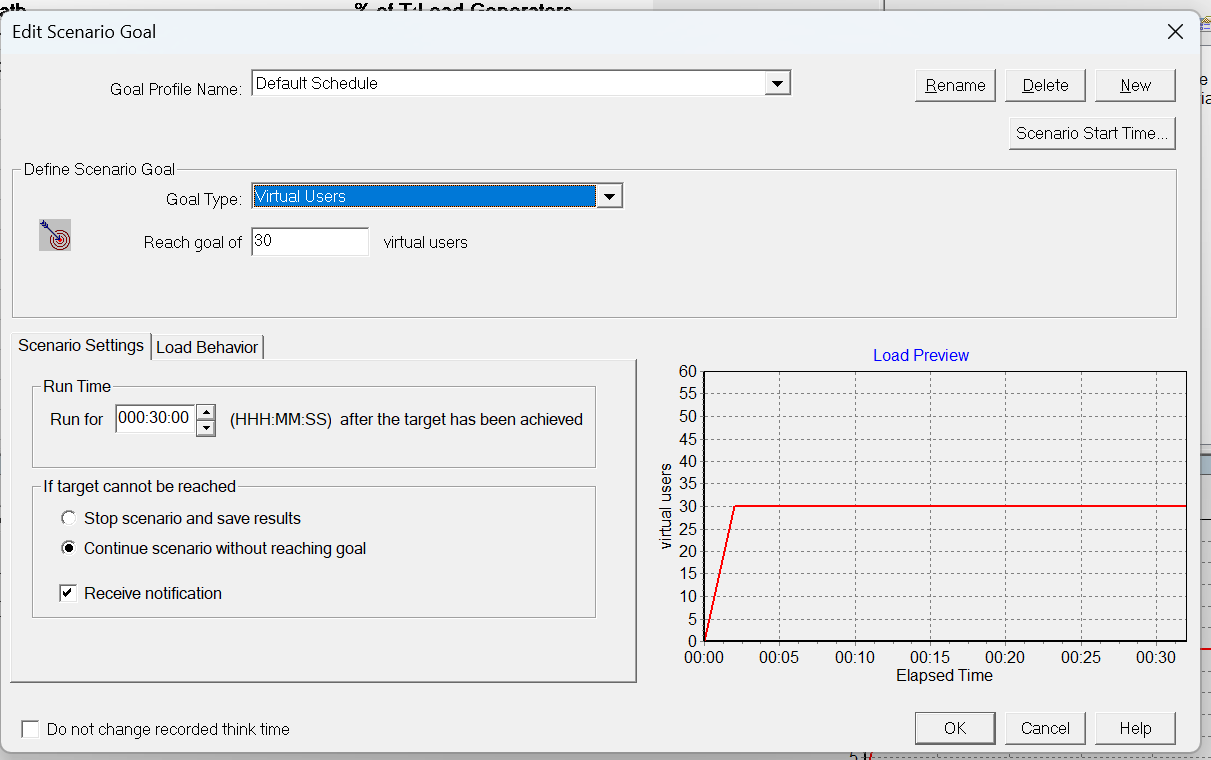
1. Run the scenario and validate the result

## Goal Oriented Scenario

In the goal-oriented scenario, you have to specify the target, and minimum and maximum no. of Vusers and duration. When the test script execution starts then LoadRunner checks whether the goal or target can be reached or not. If the target is unreachable then it stops the execution. The user load is started from min no. of Vusers to max no. of Vusers.

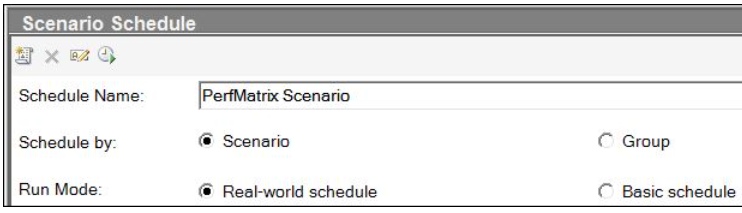
The targets can be mentioned in terms of

1. Transaction Response time
2. Virtual Load (User Load)
3. Transactions per second
4. Hits per second
5. Pages per minute



# LoadRunner – Scenario Schedule Modes

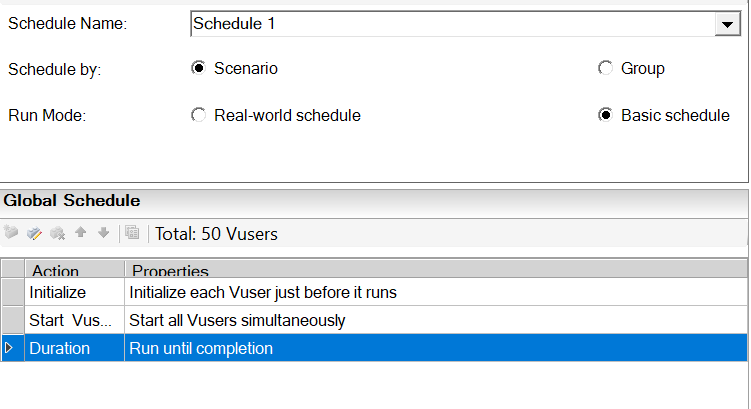
While preparing the controller scenario, the main part is the selection of the scheduling mode. The scheduling mode helps you to emulate the scenario as per your requirement. There are two options ‘Scheduled by’ and ‘Run Mode’ in LoadRunner scenario schedule modes which form four possible combinations of test scheduling mode.



## Basic Schedule

Basic schedule will run till the iterations provided in vugen.

The basic schedule is more flexible than the real-world schedule. It gives you the flexibility to have different user loads for different scripts which is not possible in the real-life schedule as it is shared equally. In the basic schedule, you have to manually pause the ramp-up whenever you want a particular user load to run whereas in the real-life schedule, this is kind of automated as you can define different ramp-up rates. You can not add, delete or modify the actions in the basic schedule mode.

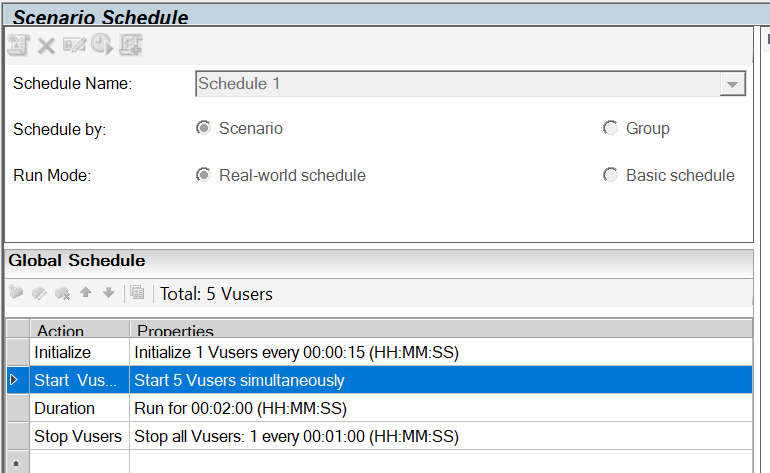


## Real Time Schedule

Real-world schedule can be used to emulate a real-life scenario where you have the load varying at different points of time. Example: If you want to ramp up 100 users first with a ramp-up of 5 users/10 sec and then after 2 mins you want to ramp in 10 more users but with a ramp-up of 10 users/10 sec. This can be done by clicking on the field marked \* in the scheduler. This kind of scenario can be emulated using a real-world schedule. In a real-world schedule, you can change the number of users by changing the Vusers value in the Start Vusers section of the Scheduler.   
Whatever number of users you define here will be equally distributed among the scripts. You can add, delete and modify the actions in the real-world scenario which is not allowed in the basic schedule.

Here, we can also provide the duration which is not possible in Basic schedule

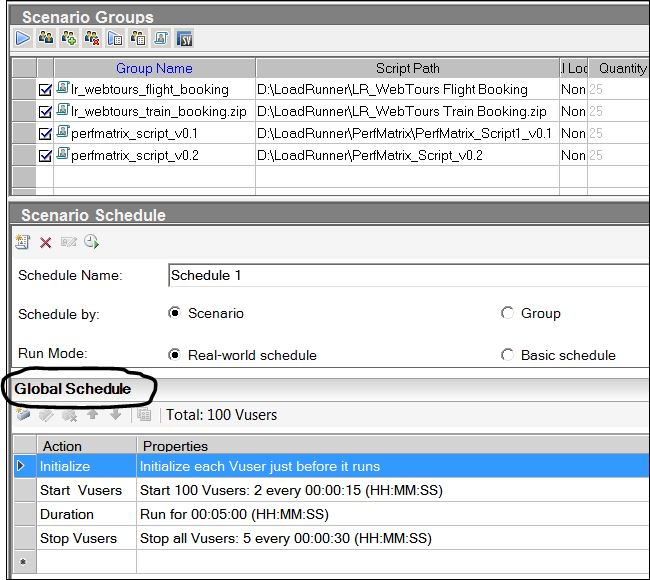
Execution will be duration dependent.



# Schedule by

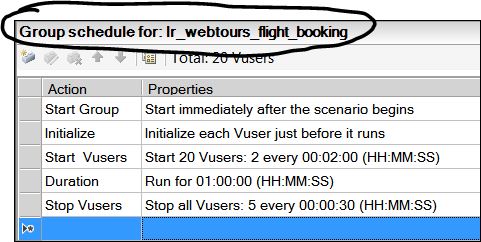
It allows the groups to run either **together**or **separate**. It has two options:

* 1. **Scenario:** *(One scenario setting applied to all groups)* When you schedule by scenario, then the scheduler settings like Vuser initialization, test duration etc. will be applied to all the groups and the controller runs all the Vuser groups participating in the scenario simultaneously. That is, the schedule defined for running the scenario is applied to all the Vuser groups. If you pay attention to the scheduler section then you can see the word “Global Schedule” which means applied to all Vuser groups.

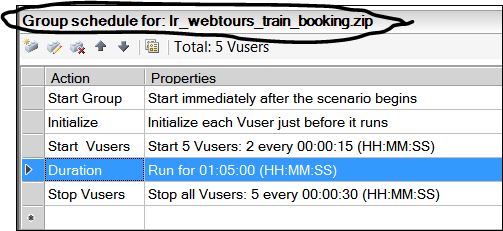


* 1. **Group:** *(Each group has separate scenario setting)* When you schedule by Vuser group, each Vuser group participating in the scenario runs on its own separate schedule. That is, for each Vuser group, you can specify when to start running the Vuser group, how many Vusers in the group to start and stop running within specified time intervals, and how long the group should continue running

Group 1:

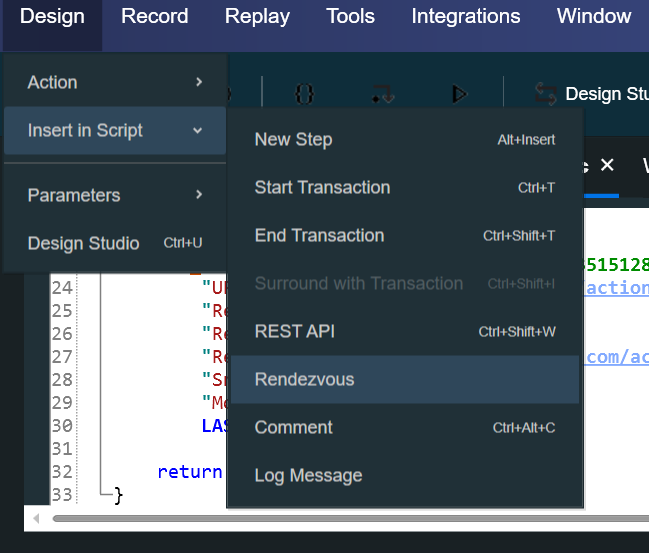
[](https://2.bp.blogspot.com/-otPkGnNVwuo/WfxiMu1MS0I/AAAAAAAADS4/avxW6nouTp8nMErd3Ujo9leF1tAjN8vgQCLcBGAs/s1600/LRS7.JPG)

Group 2:

[](https://1.bp.blogspot.com/-T0c7FuZ9G2g/WfxiMs5kfvI/AAAAAAAADS8/3ER-k_8nZ9sndbKH7GX6gHweSPdINi0-QCLcBGAs/s1600/LRS8.JPG)

# Adding Rendezvous Point

* Add Rendezvous point in Vugen.
* Place the cursor at the point when we want to insert Rendezvous point.



* Give a name to Rendezvous point
* Run from the controller to see the Rendezvous point being reached.

# Working with SLA

SLAs are specific goals that you define for your load test scenario. Analysis compares these goals against performance-related data that LoadRunner gathers and stores during the run, and then determines the SLA status (Pass or Fail) for the goal.

For example, you can define a specific goal, or threshold, for the average transaction response time measurement for any number of transactions in your script.

After the test run ends, LoadRunner compares the goals you defined against the actual recorded average transaction response times. Analysis displays the status of each defined SLA, either **Pass** or **Fail**. For example, if the actual average transaction response time did not exceed the threshold you defined, the SLA status will be **Pass**.

## Adding a SLA

To work with service level agreements (SLAs) in LoadRunner, you can:

* Define SLAs

Go to the performance scenario's design menu, click the SLAs tab, and select New to create a new SLA. You can define goals like transaction response time, errors per second, and measurements per run.

* Run the test

LoadRunner collects performance data during the test run.

* Analyze the results

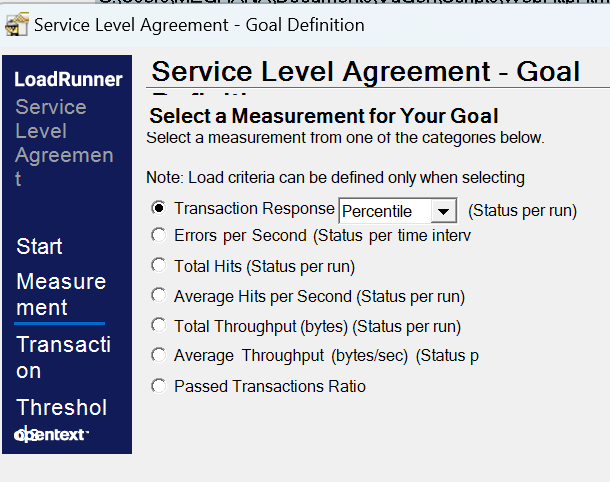
LoadRunner compares the test run data to the SLA settings to determine if the SLA passed or failed. The results are displayed in the Summary Report and the SLA Report.

* Edit SLA settings

You can edit the SLA settings after the scenario run, and the Statistics section will update to reflect the new settings.

### To add an SLA

1. Click on New in the SLA window
2. Click on Next



1. Select the SLA
2. Give the expected output
3. Click on Finish
4. View the result in the Analyser

