### What is Load Testing?

Load Testing is a type of Performance Testing that determines the performance of the system under real life conditions. It examines how the system behaves during normal and high loads and determines if a system, piece of software, or computing device can handle high loads given a high demand of end users.

### Load Testing is used to identify the following :

* The maximum operating capacity of an application
* Determine whether the current infrastructure is sufficient to run the application
* Sustainability of application with respect to peak user load
* Number of concurrent users that an application can support, and scalability to allow more users to access it.

### What is JMeter?

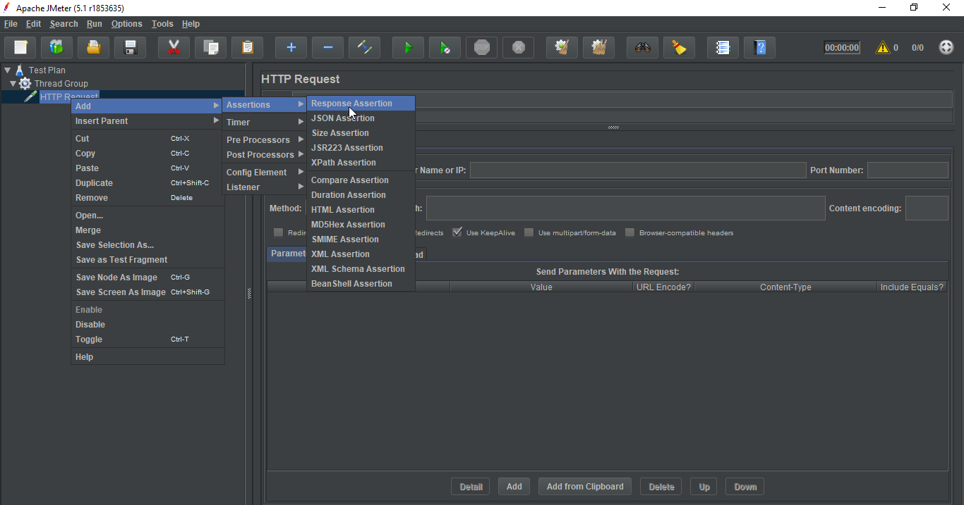
Apache JMeter is a testing tool used for analyzing and measuring the performance of different software services and products. It is a pure Java open source software used for testing Web Application or FTP application, which allows:

* **Recording** – JMeter allows users to record HTTP/HTTPS to create a Test plan using Recording facility. We use Proxy Server that allows JMeter to watch and record your actions while you browse your web application with your normal browser.
* **Reporting** – JMeter supports dashboard report generation. A host of reports are generated through JMeter which helps the user to understand Performance test execution results.

#### Elements of JMeter

The different components of JMeter are called Elements. Each Element is designed for a specific purpose. Some of the main elements are :

* **Thread Group** – Thread Groups is a collection of Threads. Each thread represents one user using the application under test. It simulates one real user request to the server. The controls for a thread group also allow you to set the number of threads for each group.
* **Samplers** – JMeter supports testing HTTP, FTP, JDBC and many more protocols.Thread Groups simulate user requests to the server. Samplers help the Thread Group to know which type of requests it needs to make.
* **Listeners** – Listeners show the results of the test execution. They can show results in a different format such as a tree, table, graph or log file.
* **Configuration** – Configuration Elements are used to set up defaults and variables for later use by samplers.
* **Assertion** – in JMeter is used to validate the response of the request, that you have sent to the server. Assertion is a process where you verify the expected result with the actual result of the request at run time. If you need to apply assertion on a particular Sampler, then add it as a child of that Sampler.

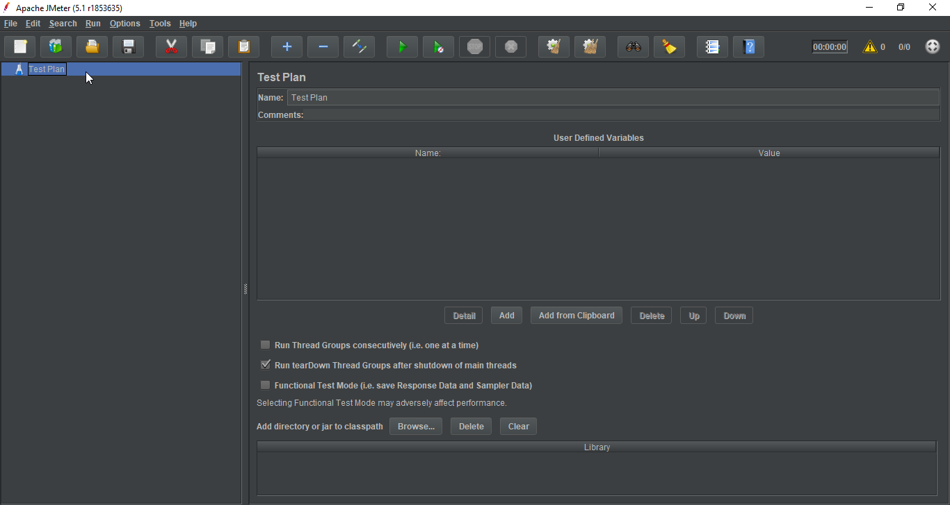


If you need to apply assertion on a particular Sampler, then add it as a child of that Sampler. You can view assertion results by adding “Assertion Listener” to the Thread Group.

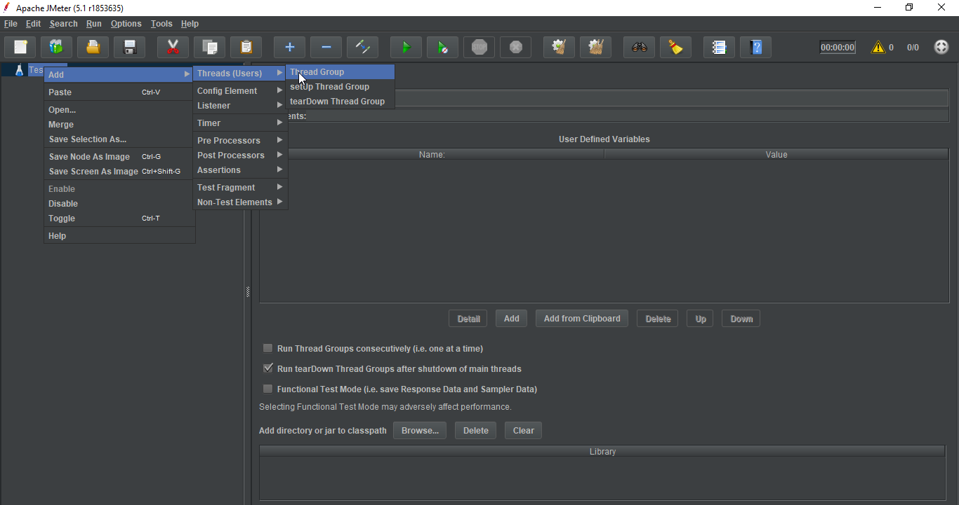
* **Response Assertion** – Response Assertion can be used to add and compare pattern strings against one or many values of server response. For Example, when you send a request to the URL: https://www.google.com and get the server response, you can verify this by using Response Assertion.

### Load testing using JMeter with the command prompt.

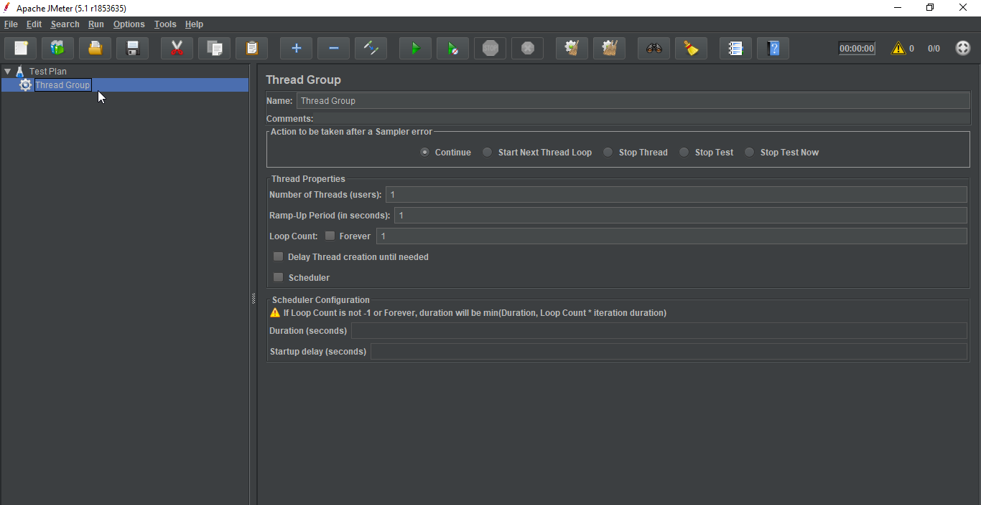
**Step 1** – First you have to create a test plan in the user interface of JMeter.



**Step 2** – Next step is to add thread groups and specify the number of threads or users and loop count.



**Step 3** – Once the thread group is created, next step is to add the HTTP Request and specify the server name and path. Now save your Test Plan in any folder.



The command to run the Test Plan in the command prompt is as follows :

* **jmeter -n -t testscript.jmx -l resultfile.csv**

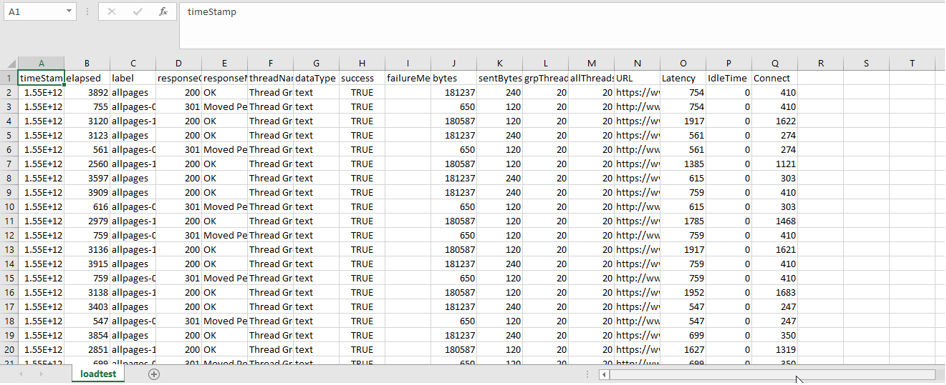
Where,

-n -> non-GUI mode

-t -> Location for jmeter test script

-l -> Location of the result file

Now once you have run the test from the command prompt, it will store the result in loadtest.csv file.



### Report Generation on Dashboard

JMeter supports dashboard report generation to get graphs and statistics from a test plan. The dashboard generator is a modular extension of JMeter. Its default behavior is to read and process samples from CSV files to generate HTML files containing graph views. It can generate the report at the end of a load test or on demand.

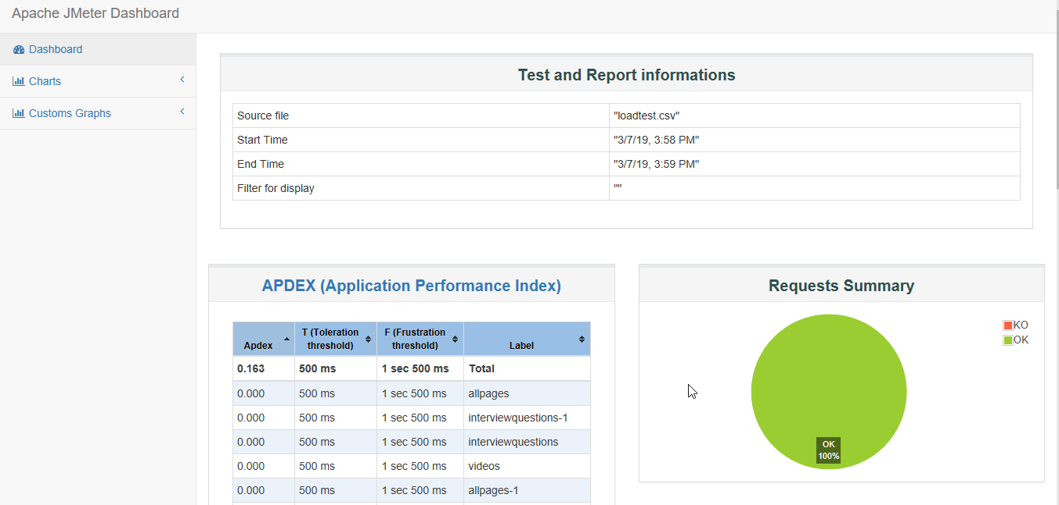
The command to generate report on Dashboard is:

* **jmeter -n -t testscript.jmx -l html.csv -e -o location**

Where,

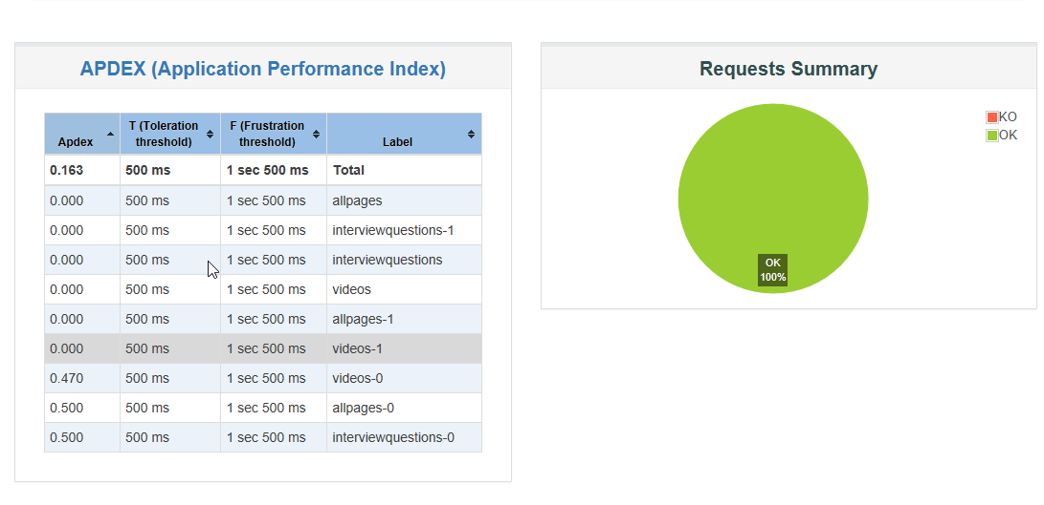
-e -> To generate HTML Reports

-o -> Location of the Output folder

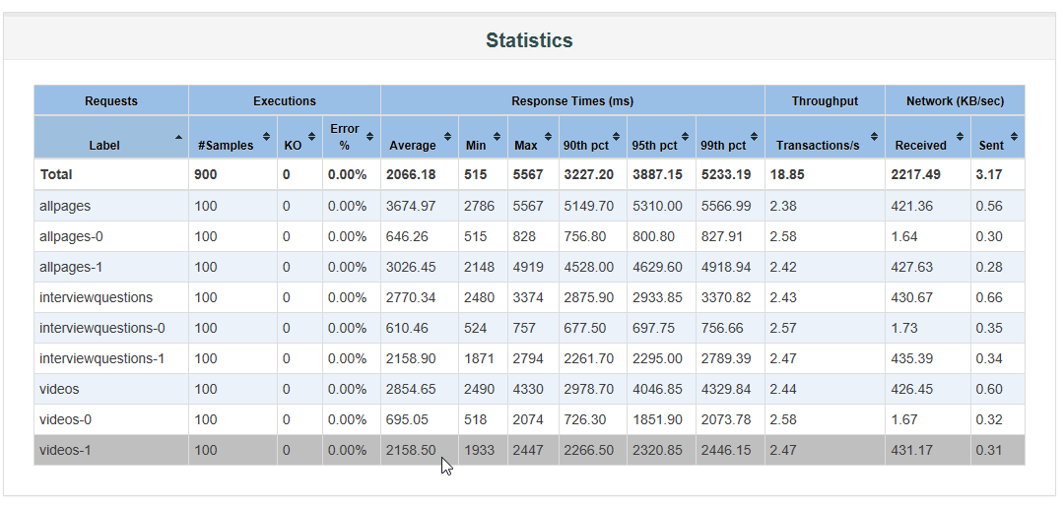


This Report provides the following metrics:

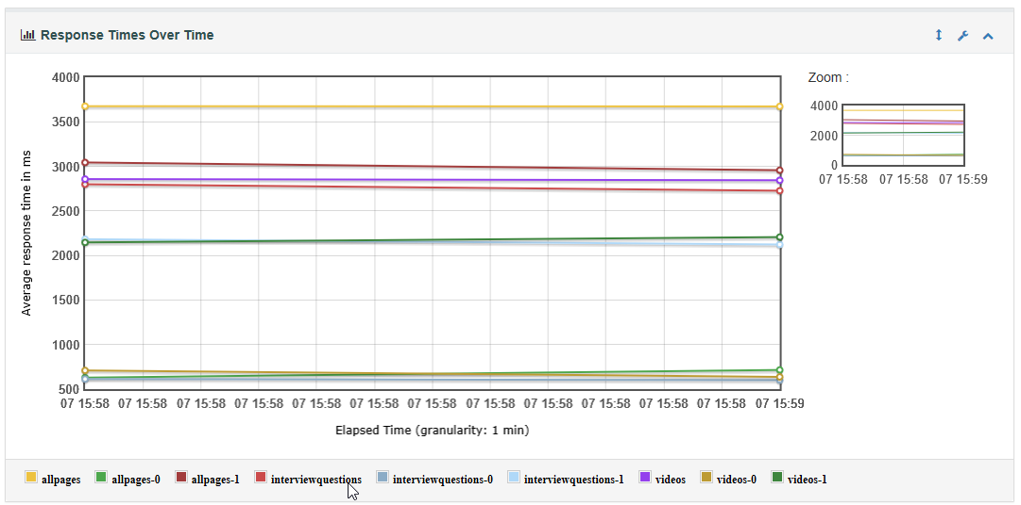
* **APDEX** (Application Performance Index) table computes for every transaction based on configurable values for tolerated and satisfied thresholds:



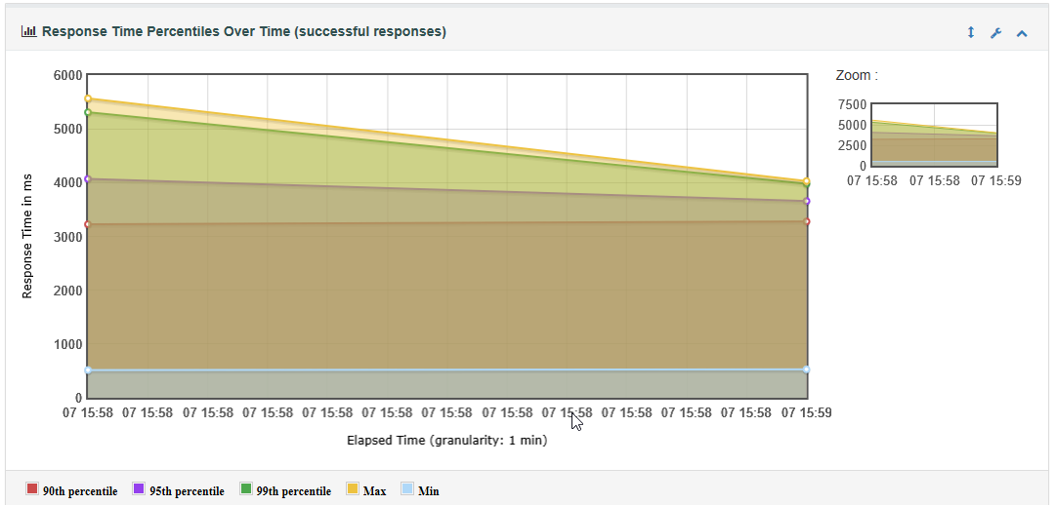
* A **Statistics table** providing in one table a summary of all metrics per transaction including 3 configurable percentiles:

.

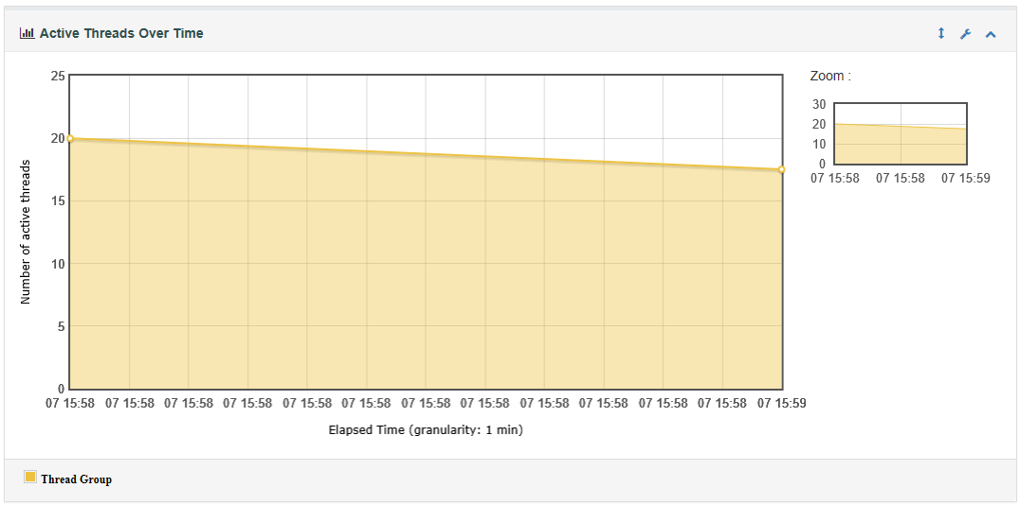
* **Response times Over Time** which includes Transaction Controller Sample Results:



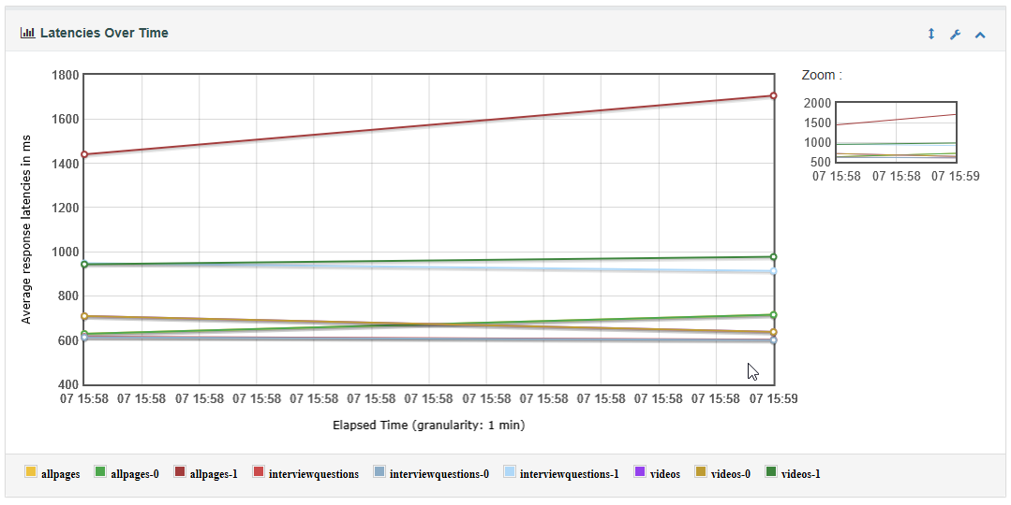
* **Response times Percentiles** **Over Time** which include successful responses only:



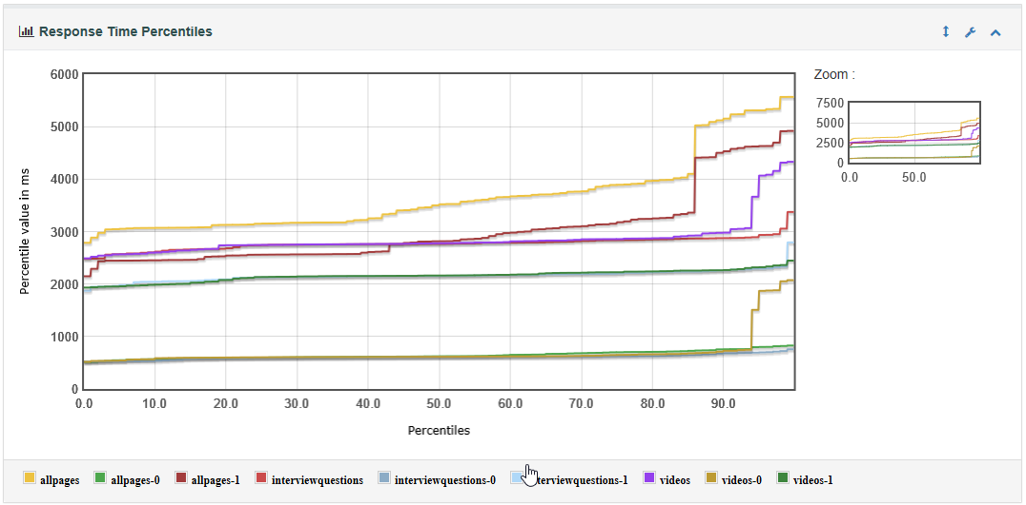
* **Active Threads Over Time** :



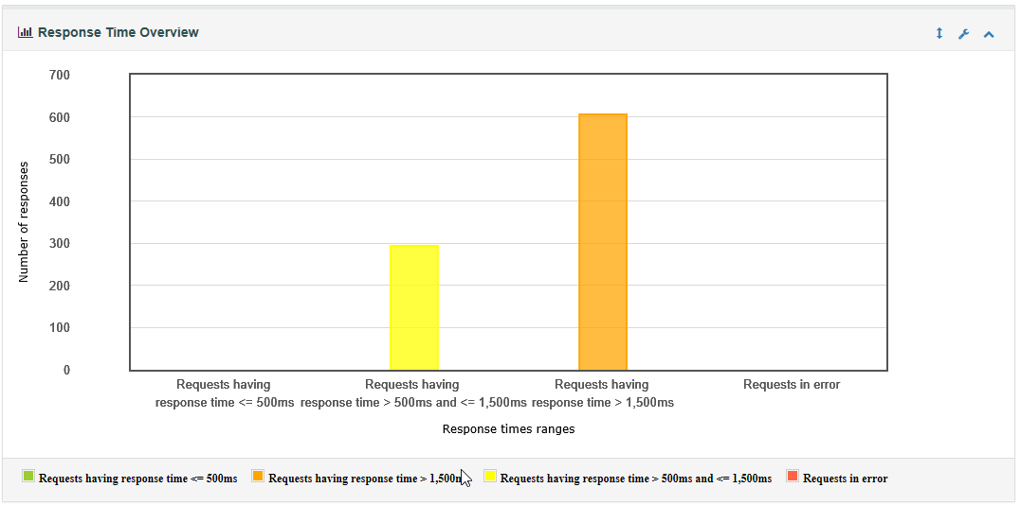
* **Latencies Over Time:**



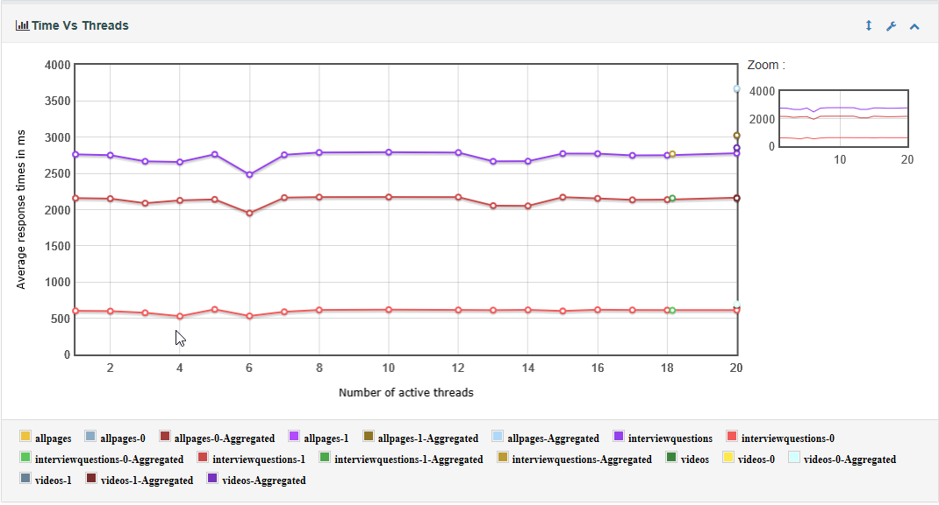
* **Response Time Percentiles** which includes Transaction Controller Sample Results:



* **Response Time Overview** which excludes Transaction Controller Sample Results:



* **Times vs Threads:**



* **Response Time Distribution:**

