

Single Waterfall Chart

To select Waterfall View, click **Waterfall** (a) on the toolbar.

By default, Waterfall includes one panel (b) to displays a chart for a selected VU (c) and iteration (d).

On the chart, the vertical axle (e) displays requests, and the horizontal axle (f) displays the timeline.

Request bars depict with a green horizontal area (g) representing server time and a blue area (h) representing network time.

To display request details, mouse over its bar. It turns yellow (i) and tool-tip displays the request name and the following timing information:

- Start - the time of issuing the request
- TTFB - the time of receiving the first byte
- TTLB - the time of receiving the last byte
- End - the time when the resource was downloaded

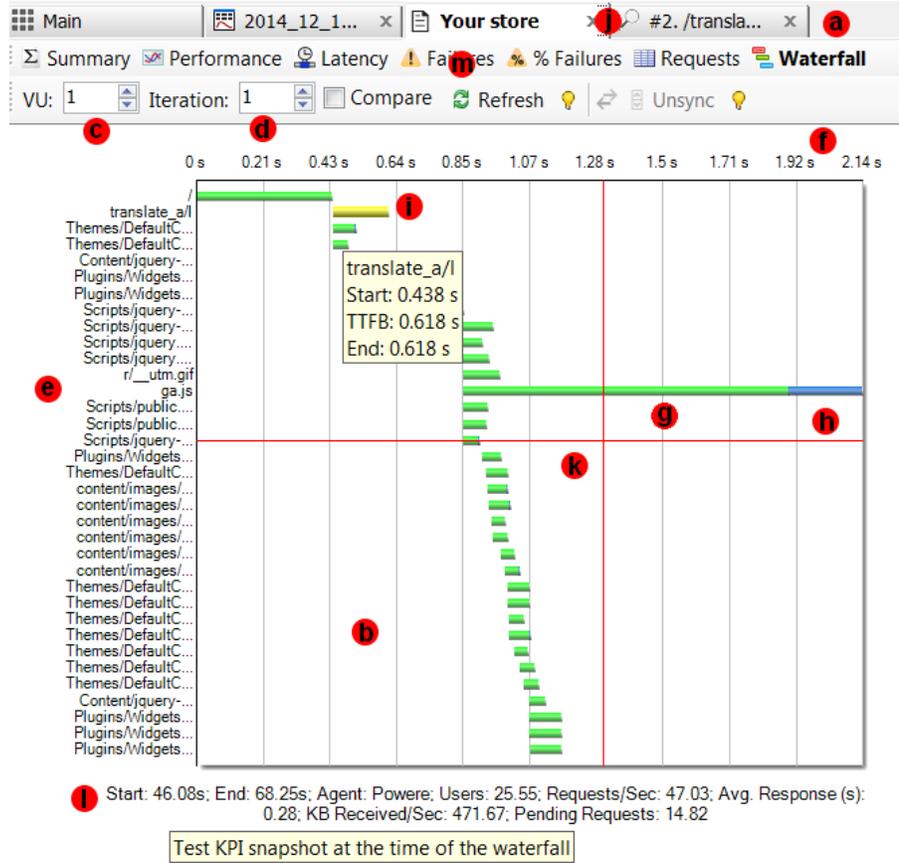
All times are measured from the moment of issuing the first request in the page /transaction.

To access the content of this request, double-click it, and a session inspector will open in a new tab (j) (see [Inspecting Sessions](#)).

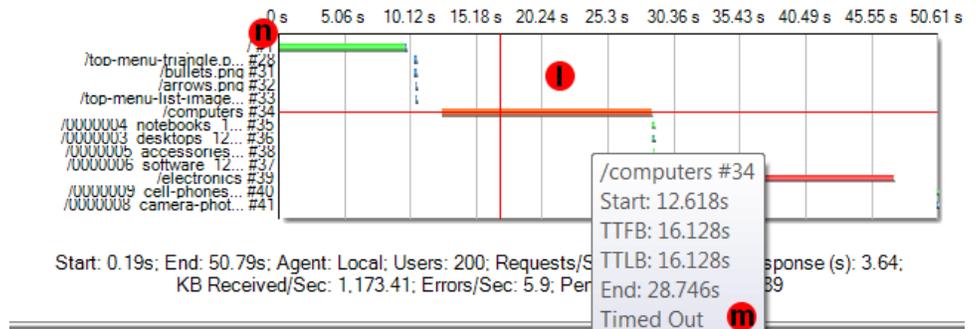
To evaluate the positioning of request /response events of page/transaction resources on the timeline, click the appropriate point in the graph panel, and a vertical and horizontal red line (k) crossing this point will appear.

Session errors and timeouts are also color-coded (l). Sessions with an error or timeout are displayed as an orange bar. The tool-tip displayed on mouseover displays its error or time out status (m).

To identify a specific request, look for its request URL or a session number (n).



Waterfall, Test Case: estore-sample.stresstimulus.com, VU: 93, Iteration: 1



To put the waterfall in the context of the load test, a key performance indicator snapshot (l) taken during the time of the waterfall is displayed on the bottom. It includes the following parameters start, End - waterfall beginning and ending timestamps on the test timeline.

- Start, End - waterfall beginning and ending timestamps on the test timeline.
- Agent - The name of the Agent where the waterfall was captured.

- **Users** - The number of instantiated and active VUs, iterating through their particular test cases. Some VUs can complete the test earlier than others and become inactive. VUs which completed all their iterations before the test end are excluded from the active user count. If the test is configured to complete only after all VUs complete their iterations, then the User graph will show a gradual declining the number of VUs at the end of the test.
- **Req/Sec** - The average number of requests being sent per second.
- **Avg. Response(s)** - The average response time of the sent requests. Every datapoint on the response time graph reflects the average value during the checkpoint intervals. If **no responses were received** between two datapoints, the response time could not be determined, so such datapoints are skipped to reflect an accurate response time curve.
- **KB Received/Sec** - The average number of bytes received per second.
- **Pending Requests** - The average number of pending requests. Pending requests are those that are issued, but StresStimulus do not yet receive the response. Generally, the greater number of pending requests indicates a slower server response. This parameter can be used to gauge performance change in several test runs.

To select a different VU / iteration, adjust information in the numeric text boxes (c) and (d). Sometimes, you may also need to click the Refresh button (m).

Note: To generate a waterfall chart, session content must be saved in storage accessible from the controller. This information is available in tests that ran from the controller without agents. In [distributed tests](#), when SQL Server is used as [storage](#), this information is also available on the controller. However, in distributed tests with SQL Server CE-based storage, the sessions initiated on the agents are stored on the agents. You still can create a waterfall chart on the controller, if **Save sessions from agents** property in **Other Options** -> **Test Result Storage** section is set to **Yes** (default) because this setting will force replicating necessary data from the agents to the controller.