Performance Testing Course : JMeter - Thread Group

Author: Hien HOANG
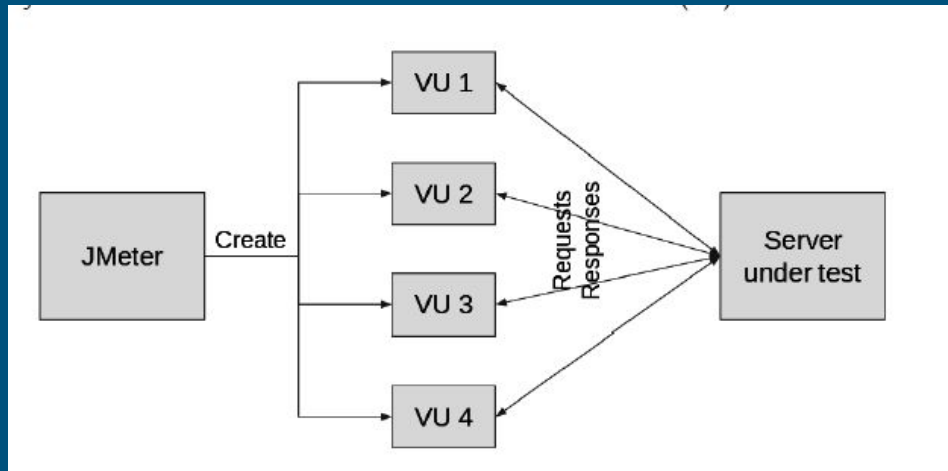
Agenda

- Introduction about each type of thread group.
- Showcase.
- How to calculate VUs

Definition

- Thread Group = pool of virtual users that execute a particular test case.
- In JMeter, we could control:
  - Number of simulated thread.
  - Ramp-up time
  - Loop count
  - Duration with start time for the test.

Basic Thread Group

It has below attributes:

- Name : Provide thread group name.
- Comments: Provide comments.
- Actions to be taken after Sampler error: Continue, Start Next Loop, Stop Thread, Stop Test and Stop Test Now.
- Number of Threads : Number of users to simulate.
- Ramp-up period: How long JMeter should take to get all the threads started.
- Loop count : Number of times to iterate the test cases. "Forever" cause the test to run until duration reaches/manual stop.
- Delay Thread creation until needed: If checked then threads are created only when the appropriate proportion of ramp-up time has elapsed.

Basic Thread Group(continued)

- Scheduler: Checked to schedule your test.
- Start time : If scheduler is checked then we could select ABSOLUTE start time. When we start the test, it will wait until the specified start time to begin the test.*(Note that Startup Delay Time will overrides start time) -> deprecated.*
- End time: If scheduler is checked then we could select ABSOLUTE end time. When we start the test, it will wait until the specified start time to begin the test and stop at specified end time.*(Note that Duration will overrides end time) -> deprecated.*
- Duration : Denotes total duration of the test.
- Start-up delay: How long JMeter should wait before starting the test.

Basic Thread Group(continued)

Showcase:

- Modify the script basicTG_1.jmx to understand each properties.

Pre-requisites : Plugin manager

How to download?

- Go to https://jmeter-plugins.org/install/Install/
- Download plugin-manager.jar and place it into lib/ext folder.
- Restart JMeter.
- Click on Options > Plugin Manager.
- Select these following plugins : 3 Basics Graphs, 5 Additional Graphs, Custom Thread Groups, Distribution/Percentiles Graphs

setUp Thread Group

- Special type of thread group that can be used to perform pre-test actions, such as : connection to the database, etc…
- Same attribute as basic thread group

Showcase:

- Use setUp Thread group to generate file csv containing all contacts from https://restool-sample-app.herokuapp.com/#/characters
- Use the csv file in Thread group to print values.

tearDown Thread Group

- Special type of thread group that can be used to perform post-test actions, such as : connection to the database, etc…
- Same attribute as basic thread group

Note : By default, tearDown thread group won't run if the test is completed as expected. If you want to make it run anyways, so check on "Run tearDown Thread Group after the shutdown of main threads."

Showcase:

- Add tearDown group on the previous test to delete csv file created in previous step.

Ultimate Thread Group

- *Name*: To provide Thread Group name. It may be your Test Case/Test Scenario name
- *Comments*: To provide comments
- *Action to be taken after a Sampler error*: Actions taken in case of sampler error occurs, either because the sample itself failed or an assertion failed etc. The possible choices are: Continue, Start Next Loop,Stop Thread,Stop Test, Stop Test Now
- *Thread Schedule*: Number of users to simulate.
- *Start Thread Count*: The count of threads for a particular time slot.
- *Initial Delay*: Initial delay is the time for the first user (thread) to hit the server after JMeter starts. This time is nothing but just after how many seconds you want to hit the server first time.
- *Start-up time*: Start up represents ramp-up time and it divides among each user.
- *Hold Load For*: It represents a steady state of workload scenario.
- *Shutdown Time*: It represents ramp-down time. same concept as Start-up time.

Ultimate Thread Group(continued)

Showcase:

- Modify the script UTG-Stepping.jmx for each properties.
- Verify result with jp@gc - Active Threads Over Times.

Concurrency Thread Group

- Name : Same as Basic TG.
- Comments : Same as Basic TG.
- Action to be taken after a sampler error : Same as basic TG.
- Target Concurrency: Total number of threads in the test.
- Ramp-up Time: Total ramp-up duration in minutes.
- Ramp-up Steps Count: Total number of steps in which a group of threads will ramp-up.
- Hold target rate time : Steady state when all threads are active.
- Time Unit: Change input time unit either in minute or in second.
- Thread Iterations Limit: Purpose of this option could be control loop count, but it does not work.
- Log Threads Status into File : Provide full path of file along with the name where you want to save log file.

Concurrency Thread Group(continued)

- Modify the concurrencyTG.jmx to understand each properties.
- Run and verify against jp@gc - Active Threads Over Times.

Arrival Thread Group

- Attributes:
  - Name : Same as Basic TG.
  - Comments : Same as Basic TG.
  - Action to be taken after a sampler error : Same as basic TG.
  - Target rate (arrivals/min): Total number of iterations to be started in 1 min.We can also provide target rate in seconds.
  - Ramp-up Time: Total ramp-up duration of arrival rate in minutes.Can also select in seconds.
  - Ramp-up Steps Count: Total number of steps in which arrival count will ramp-up.
  - Hold Target Rate Time(min): Steady state where all threads are active.
  - Time Unit: Change input time unit either in minute or second
  - Thread Iterations Limit: Defined number of iterations completed by each thread. If blank then thread will continue the iteration until the test ends.
  - Log Threads Status into File : Provide full path of file along with the name where you want to save log file.

Arrival Thread Group(continued)

- <u>Showcase:</u>
  - Modify the arrivalTG.jmx to simulate the case when 500 purchasing requests are created in 3 minutes.

Free-form Arrival Thread Group

- More flexible than Arrival Thread Group: Can do spike test.

=> Can apply conditional load on the server.

Free-form Arrival Thread Group(continued)

Attribute:

- Name : Same as basic TG.
- Comments: Same as basic TG.
- Actions to be taken after a Sampler error: Same as TG.
- Thread Schedule(arrivals/min) : Total number of iterations to be started in 1 minute. Start Value, End Value and Duration are compulsory.
- Add Row: Add new row in Thread Schedule section.
- Copy Row: Copy the selected row in Thread Schedule section.
- Delete Row: Delete the selected row in Thread Schedule section.
- Time Unit : Change input time unit either in min or second.
- Thread Iterations Limit: same as Arrival TG.
- Log Thread Status into File : same as Arrival TG.

Free-form Arrival Thread Group(continued)

Modify freefromArrivalTG.jmx to simulate the case when in in normal case, 100 purchase requests are generated in 3 minutes, but when event happened, 200 purchase requests carts are processed in 1 minute.

How many VUs do I need?

1. Identify critical business scenarios to script:
    - If business scenarios are not well defined, tests will not conform to what will happen in production -> Tests become useless or even dangerous, giving false impression.
    - To define these scenarios:
        - Retrieve statistics of application if they exist.
        - Define with Users/Product Owner.
    - It is advisable to apply Pareto Principle and therefore to take (at least as first step) 20% of scenarios that cover 80% of the scope of application.
    - Make risk study to prioritize the tests because this allow to choose tests in case of tight schedule.

NOTE: Can select scenario based on : Frequency, Vitality and Risky.

How many VUs do I need?(continued)

2. Calculation no. of VUs:

    ● Maximum number of VUS configurable per injector depends on :
        ○ Test plan complexity.
        ○ Number of listeners and their nature.
        ○ JMeter execution type.
        ○ Resources available for the test(servers,processor,memory,etc...)
    ● But target load is not necessary to equal to maximum number => 2 strategies: 1 VUs = 1 user and 1 VU = n users.
    ● For second strategy, we could use the calculator provided by https://ubikloadpack.com/virtual-users-computer.php

NOTE: Do not forget to take into account requests made by bots.