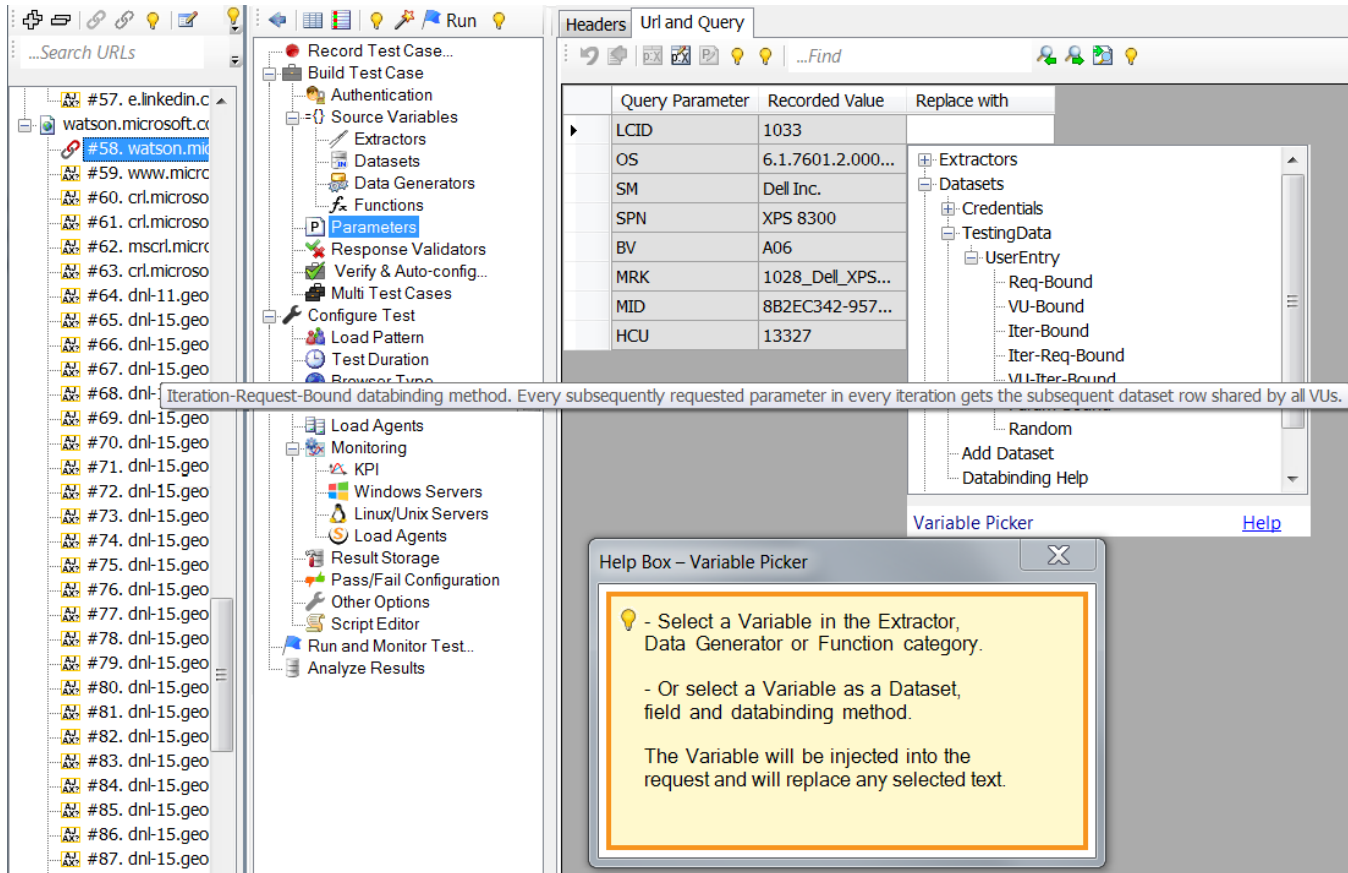


Databinding

Databinding is a rule for fetching dataset records used for parameterizing requests. The dataset stores multiple sequential records. Every time a dynamic request with a parameter bound to the dataset is generated, a single dataset record is consumed. In some tests, it is necessary to use different records every time by repeatedly looping through the records in the dataset. However, in other tests, it is required that a record is repeated for either the same VU or requests or iterations or combinations thereof. For example, in login requests, a VU has to use the same credentials stored in the dataset on all requests and all iterations. At the same time, to realistically load test a data entry scenario, a VU needs to use different dataset records.

StresStimulus supports seven databinding methods shown below. They appear in the variable picker control under the selected dataset field. To display the description of a data binding method, mouse over it.



Note

If the dataset has Hide Advanced Databinding selection set to Yes, then the databinding methods will not be displayed. For more, click [here](#).

A description of the databinding methods is provided in the table below. The following examples show which dataset record is used in each method: The sample dataset has 20 records. The test case includes five parameterized requests bound to this dataset. Two VUs run the test through two iterations. Both VUs traverse through their respective iterations asynchronously as they emulate independent physical users, so there is no synchronization between requests issued by different users.

Description	Example
-------------	---------

1. Request-Bound data-binding method (Req-Bound).

Every parameter requested by any VU in any iteration gets a subsequent dataset row.

	VU1	VU2	VU1	VU2
R eq.	Iter . 1	Iter . 1	Iter . 2	Iter . 2
1	1	2	10	13
2	3	4	12	14
3	5	7	17	15
4	6	9	18	16
5	8	11	19	20

2. VU-Bound data-binding method (VU-Bound).


Every VU gets a subsequent dataset row used for all its parameters requested in all iterations.

	VU1	VU2	VU1	VU2
R eq.	Iter . 1	Iter . 1	Iter . 2	Iter . 2
1	1	2	1	2
2	1	2	1	2
3	1	2	1	2
4	1	2	1	2
5	1	2	1	2

3. Iteration-Bound data-binding method (Iter-Bound).

Every iteration gets a subsequent dataset row used by all VUs in all requested parameters.

Note:

 If the parameter is inside a loop object like [Loop](#) or [Do...While](#) with **Increment the index for iteration-bound datasets** set to yes, then on every loop, the record will be selected by the same rule.

See below for more info.

	VU1	VU2	VU1	VU2
R eq.	Iter . 1	Iter . 1	Iter . 2	Iter . 2
1	1	1	2	2
2	1	1	2	2
3	1	1	2	2
4	1	1	2	2
5	1	1	2	2

4. Iteration-Request-Bound data-binding method (Iter-Req-Bound).


Every subsequently requested parameter in every iteration gets the subsequent dataset row shared by all VUs.

	VU1	VU2	VU1	VU2
R eq.	Iter . 1	Iter . 1	Iter . 2	Iter . 2
1	1	1	6	6
2	2	2	7	7
3	3	3	8	8
4	4	4	9	9
5	5	5	10	10

5. VU-Iteration data-binding method (VU-Iter-Bound).

Every VU on every iteration gets a unique value. The value is the same for all requests within the iteration.

Note:

 If the parameter is inside a loop object like [Loop](#) or [Do...While](#) with **Increment the index for iteration-bound datasets** set to yes, then on every loop, the record will be selected by the same rule.

See below for more info.

	VU1	VU2	VU1	VU2
R eq.	Iter . 1	Iter . 1	Iter . 2	Iter . 2
1	1	2	3	4
2	1	2	3	4
3	1	2	3	4
4	1	2	3	4
5	1	2	3	4

6. Parameter-Bound data-binding method (Param-Bound).

Every requested parameter gets a subsequent dataset row shared by all VUs in all iterations.



	VU1	VU2	VU1	VU2
R eq.	Iter . 1	Iter . 1	Iter . 2	Iter . 2
1	1	1	1	1
2	2	2	2	2
3	3	3	3	3
4	4	4	4	4
5	5	5	5	5

7. VU-Iteration-Random data-binding method (VU-Iter-Random). Every VU on every iteration gets a random dataset row that can be used to parameterize any request. It is somewhat similar to VU-Iter-Bound, but the rows are consumed not sequentially but in random order.	
8. Random data-binding method (Random). Every request parameter gets an arbitrary dataset row.	

Note: If the number of records in a dataset is insufficient, then the records will be reused according to the round-robin algorithm.

Databinding formulas

Below are the system formulas used to determine which row to fetch during some databinding methods

Databinding mode	Formula
VU-Bound	VU # Example: on VU #8 Iteration #6 Request #20 will use the 8th row
Iteration-Bound	Iteration # Example: on VU #8 Iteration #6 Request #20 will use the 6th row Note  If the parameter is inside a loop or do...while with Increment the index for iteration-bound datasets set to yes, then the formula becomes Loop # (number of loop iterations this parameter was in) Example: on VU #8 Iteration #6 Request #20 and every iteration has 3 loops, then will use 16th row on the first loop, 17th on the second, and 18th on the third loop iteration
VU-Iteration-Bound	$(\text{Iteration \#}) * (\text{Max VUs}) + (\text{VU \#})$ Example: with max 10VUs on VU #8 Iteration #6 Request #20 will use the 68th row Note  If the parameter is inside a loop or do...while with Increment the index for iteration-bound datasets set to yes, then the formula becomes $(\text{Loop \#}) * (\text{Max VUs}) + (\text{VU \#})$ (Loop # is described above) Example: with max 10VUs on VU #8 Iteration #6 Request #20 and every iteration has 3 loops, then will use 168th row on the first loop, 178th on the second, and 189th on the third loop iteration