# Check host’s CPU Configuration: Application to host resizes

## Contents

## Introduction

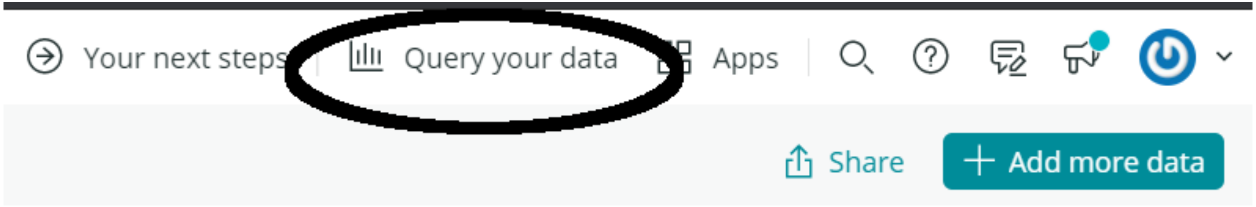
When a host undergoes a resize, the following cases may occur:

- A CPU increase/decrease of hosts existing instances (vertical scaling)
- An increase or decrease of the number of instances (horizontal scaling)
- Both of the above

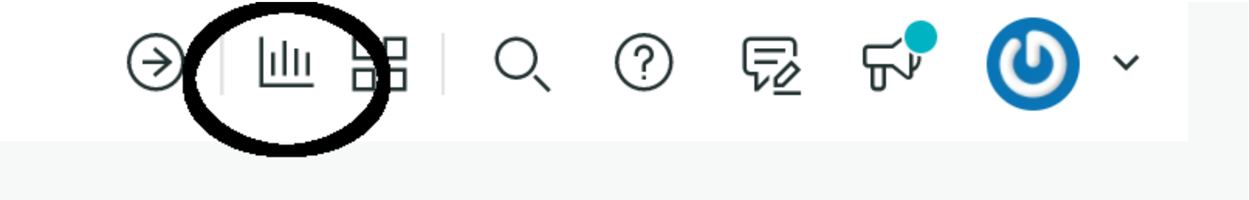
New Relic UI is capable of tracking the host’s CPU Configuration. It records/stores into the New Relic tables all resize cases and their time of occurrence. In this article, we present a set of steps and queries the user can follow, to check a host’s CPU configuration. The user may apply these steps to identify a resize, pinpoint the time of change, and view all hardware configuration changes that occurred, if any, at any point in time before or after the resize.

## User’s first steps to navigate into NR Queries

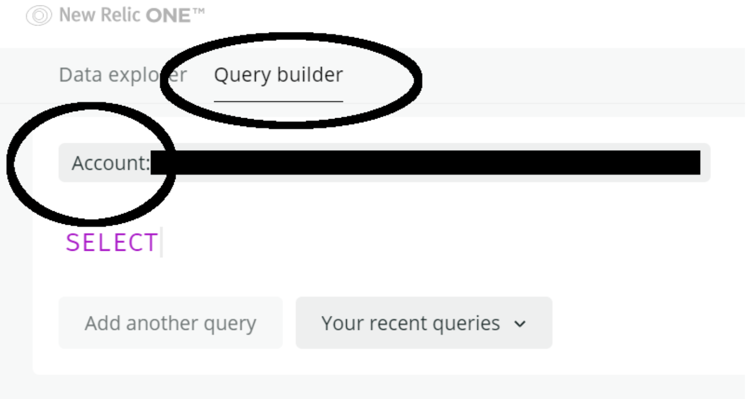
- When New Relic One log in is complete, click on the “Query your data” button on the upper right corner:



Sometimes, depending on the zoom used on the browser, the query data button does not have a written description.



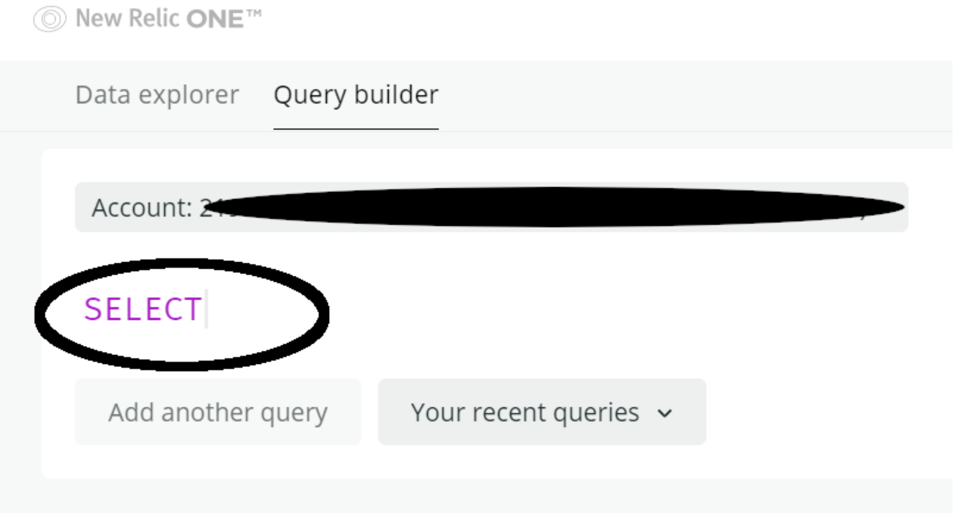
- In the upper left corner, if not selected, click on “Query builder” button. Then make certain that the account name and info shown in the drop-down menu is the one you wish to study:



Account: [redacted]

SELECT

- Now the user is ready to use the query Line (starts with the “SELECT” command)



Account: [redacted]

SELECT

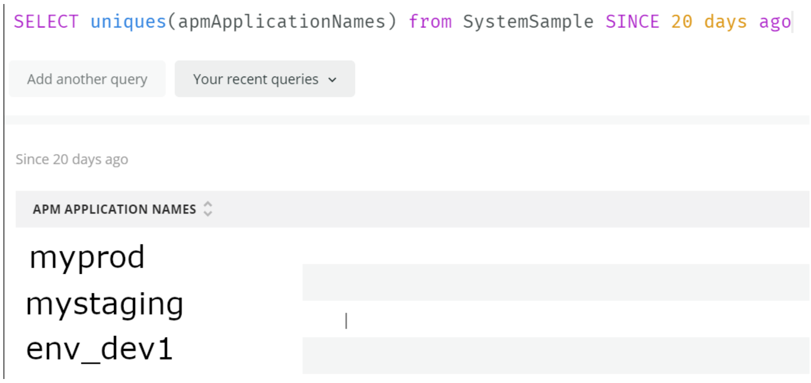
## Find the host’s environment names, choose the environment of study

To find the host’s environment’s names, one uses the following query:

```
SELECT uniques(apmApplicationNames) from SystemSample SINCE 20 days
ago
```

The timeframe defined by the keyword “SINCE” can be any timeframe the user wishes to study. It could be just a few minutes, or a number of months.

From the list of names, the user can choose the environment he/she wishes to study. Let's assume that the desired environment is the host's production, which goes under the name "myprod". This name will be used throughout all subsequent queries:



```
SELECT uniques(apmApplicationNames) from SystemSample SINCE 20 days ago
```

myprod

mystaging

env_dev1

It is the user's responsibility to find the environments name, or names, that is consistent with the time period of study. For example, let's assume that an environment of interest (it can be the Production, or any other environment) undergoes a change of names during a time period of 4 months and the user wants to observe configuration during this same time period. In this case, the user should separate the total time period of interest into sections where the environment has only one unique name, for example the first 1 month where the environment has unique name "A", and the remaining three months where the environment has unique name "B".

## Identify a resize

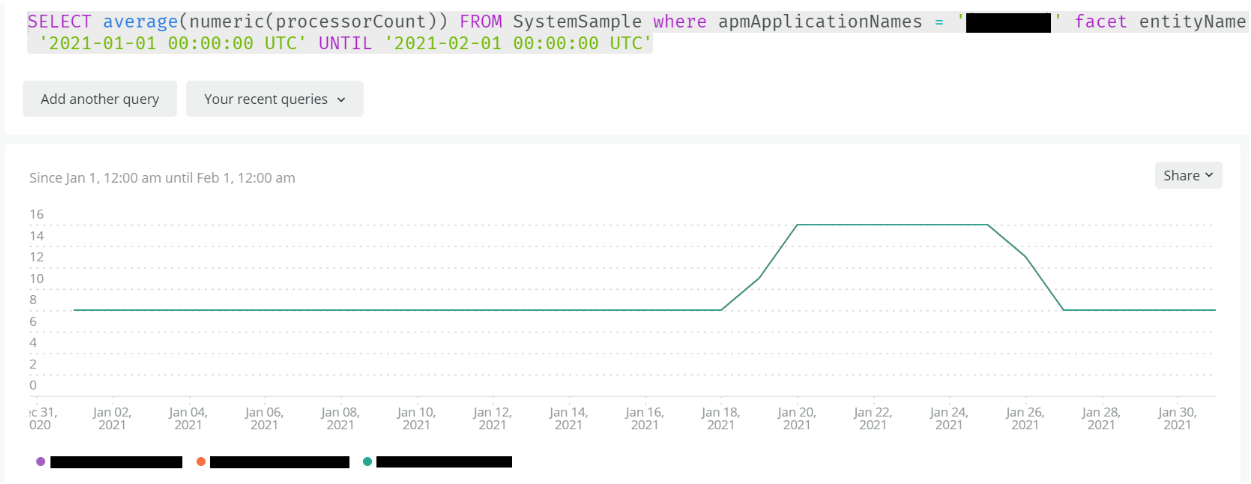
To identify a resize, the user may use the following query:

```
SELECT average(numeric(processorCount)) FROM SystemSample where apmApplicationNames = 'myprod' facet entityName timeseries 1 day limit 20 SINCE '2021-01-01 00:00:00 UTC' UNTIL '2021-02-01 00:00:00 UTC'
```

For this query, we used:

- The variable "processorCount, which provides – after the operations of the functions "numeric" and "average" - the CPU configuration of each instance.
- The NR table SystemSample, where all these variables are stored and can be retrieved.
- The "myprod" as the name of the environment of study, in this case the Production.
- A time step of 1 day (defined with the "timeseries" keyword, highlighted in green). The environments name "myprod" is used, to separate the environment the user wishes to study from the rest.
- The "limit" keyword, highlighted in gray, is used to increase the total number of instances New Relic reports. In this case, New Relic will report up to 20 instances.
- A timeframe of 1 month:1/1/2021 – 2/1/2021, defined by the "SINCE" and "UNTIL" keywords, (highlighted in yellow).

The result shown is a chart which looks like the following printscreen:



```
SELECT average(numeric(processorCount)) FROM SystemSample where apmApplicationNames = '[redacted]' facet entityName
 '2021-01-01 00:00:00 UTC' UNTIL '2021-02-01 00:00:00 UTC'
```

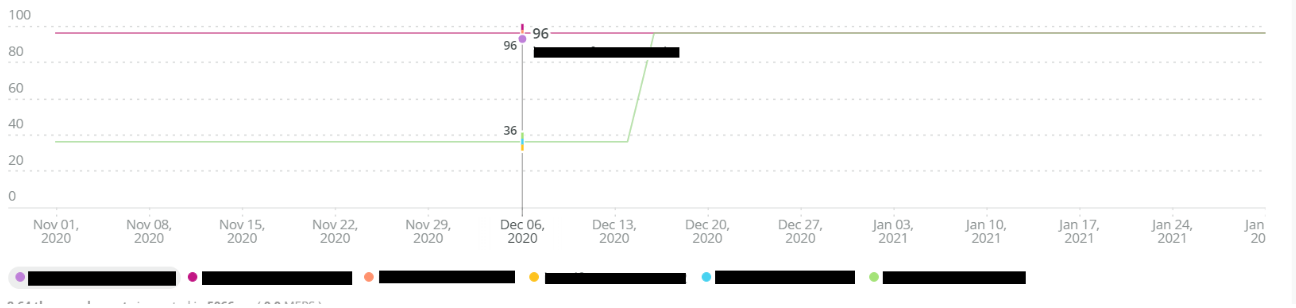
In this example, the chart covers the timeframe of 1 month with a 1-day increment, defined by the "timeseries" timestep. Therefore, the chart "covers" the timeframe of interest by showing 31 "points" (or buckets, which is how these points are named in New Relic), each point being 1 day defined by the time step. If the user does a "mouse-over" on the chart, he/she can view details such as the exact date of the point and its value.

From the chart, the user can see the following info: a) in the lower left corner, the name and number of the environment's instances b) from the chart, the date (in a 1-day timestep) where a resize had occurred. From the above chart we see three instances of the same CPU size that undergo two resizes, one upsize on Jan 19 2021 and a subsequent downsize on Jan 26 2021.

If the user wants to track down the "history" of each instance separately, whose names are indicated on the lower left corner of the chart, he/she should click once on the instance of interest to highlight it, relative to the rest. After the click, New Relic would show only the instance of interest, making the rest "invisible".

If a horizontal resize was occurring, the chart would show the instance that is added or removed with its line starting or stopping at the date of the change. For example, if a hypothetical instance i-001 was added into the host's configuration on Jan 19, then the line of this instance would start on that same day on the chart, while the other "existing" instances would have their lines started at the beginning of the timeframe.

Finally, if the host had instances of various sizes, their CPU size would show on the chart as a separate set of lines. For example, if the user performs a query on a host whose production environment has 6 instances with different sizes, the chart would look like the following printscreen:



From the above printscreen we observe that a number of instances have a CPU size of 36, while others have CPU size of 96. To find the exact configuration and values, the user can click on each instance in the lower left corner of the chart (as mentioned earlier). The click will highlight the instance in question, and a "mouse-over" will show the exact value. In the above printscreen, 3 instances are of CPU size 36, the other

3 of size 96. Around December 15, the set of instances which were on a CPU size of 36 undergo an upsize and they also become CPU size 96.

## Finetune the occurrence of a resize

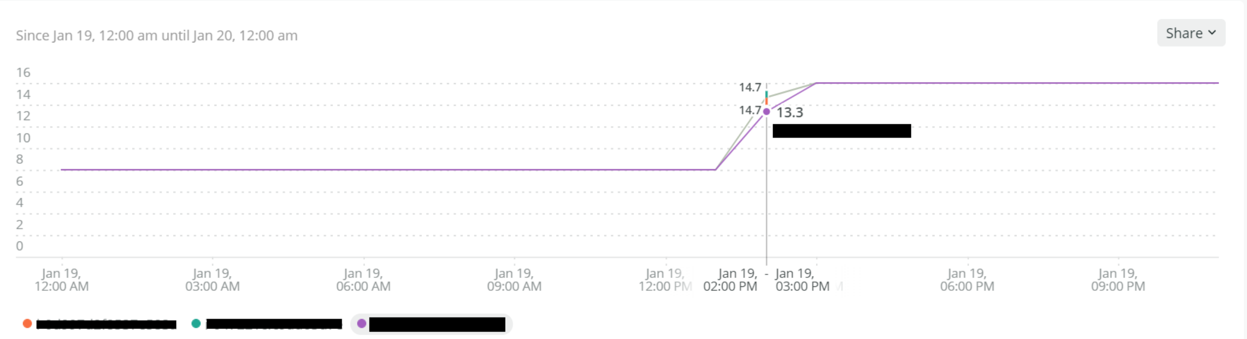
In the previous examples, the user is able to observe the CPU size configuration of the host of choice, on a day to day step. The user has the ability to refine the search, to pinpoint to the hour or minute when a resize has occurred. Once the day of the resize is identified, the user can perform queries on timeframes with higher time resolution (minutes and/or hours instead of days).

### By hour

To pinpoint the occurrence of the resize on the previous 3 instance configuration to the hour, after the day of the resize has been identified (in this case Jan 19 2021) the following query may be used:

```
SELECT   average(numeric(processorCount))   FROM   SystemSample   where
apmApplicationNames = 'myprod' facet entityName timeseries 1 hour limit
20 SINCE '2021-01-19 00:00:00 UTC' UNTIL '2021-01-20 00:00:00 UTC'
```

On the above New Relic query, we use as total timeframe the date of Jan 19 2021, which was identified by using the query of the previous chapter. Now the timeseries step is 1 hour.
The resulting chart plots the CPU size of the host's instance on a 1 hour timestep. It looks like the following:



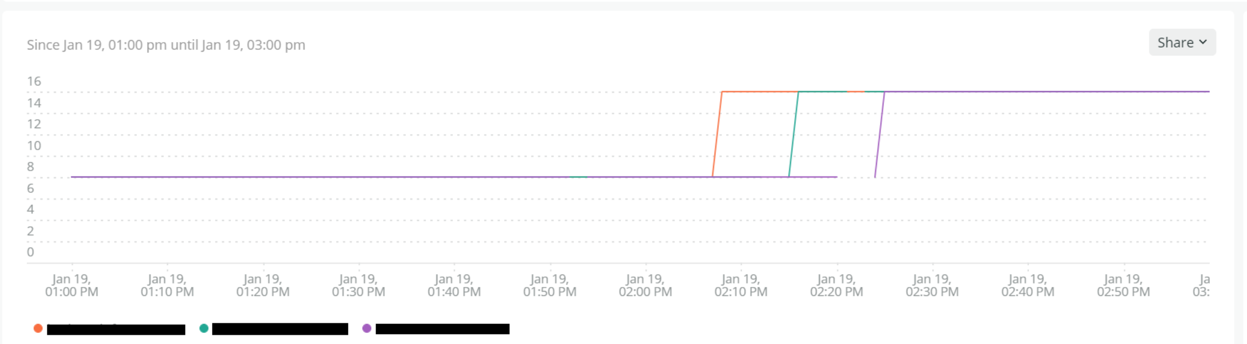
From the chart, we observe by "mouse-over" that an upsize is taking place on Jan 19 between 2:00 – 3:00 pm UTC.

### By minute

If the user wants to further pinpoint the upsize to a minute accuracy, the following query may be used which has a total timeframe of 2 hours (13-15 pm UTC) and a time step of 1 minute:

```
SELECT   average(numeric(processorCount))   FROM   SystemSample   where
apmApplicationNames  =  'myprod'  facet  entityName  timeseries  1  minute
limit 20 SINCE '2021-01-19 13:00:00 UTC' UNTIL '2021-01-19 15:00:00 UTC'
```

And the resulting chart looks like the following:



From the above chart, we make the following observation: Not all instances had been upsized at exactly the same time. Some had taken an additional 20 minutes to be resized, relative to the ones that were resized first.