**EPA CA2**

X00130180 – Jordan Williams

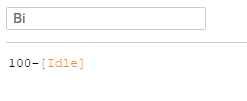
Script: <runtest.sh>

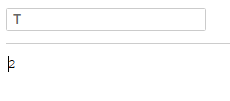
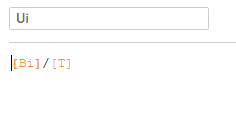
Data Visualisation

After I had gathered all the necessary data it was then time to begin the visualisation outlined in the document.

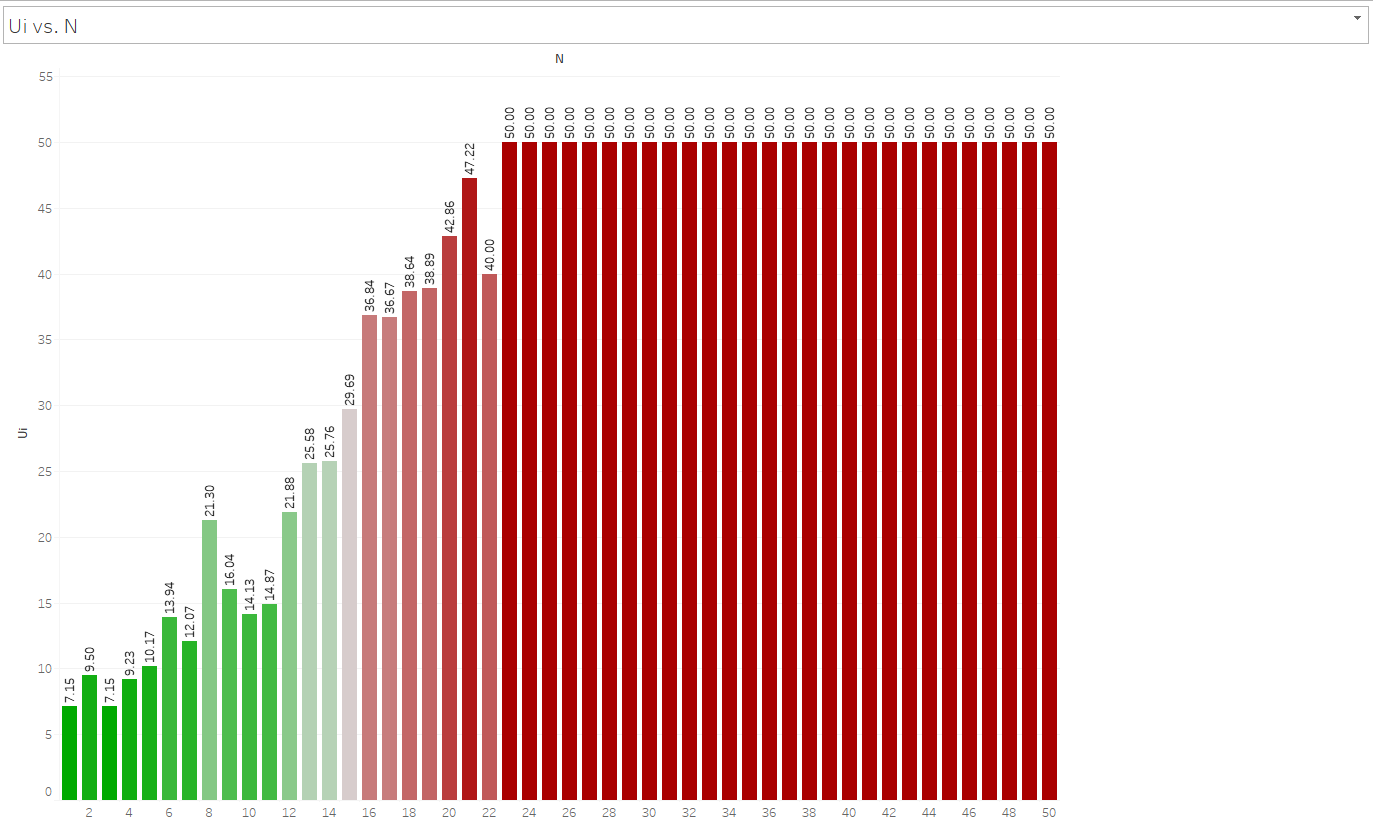
**Ui vs. N**

I went about calculating this was via the following method:





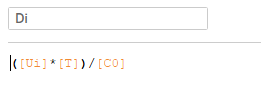
That left me with the following result of:



As you can see the the utilisation of resources is consistently rising the longer it goes, until eventually it gets maxed out and no longer has anymore resources to devote to this task. This explains why from N=24 Ui results in 50, the max. The results weren’t surprising as I was only running my machine with limited resources, such as 1 cpu. With such limited resources, its easy to see why the machine quickly utilised all its resources.

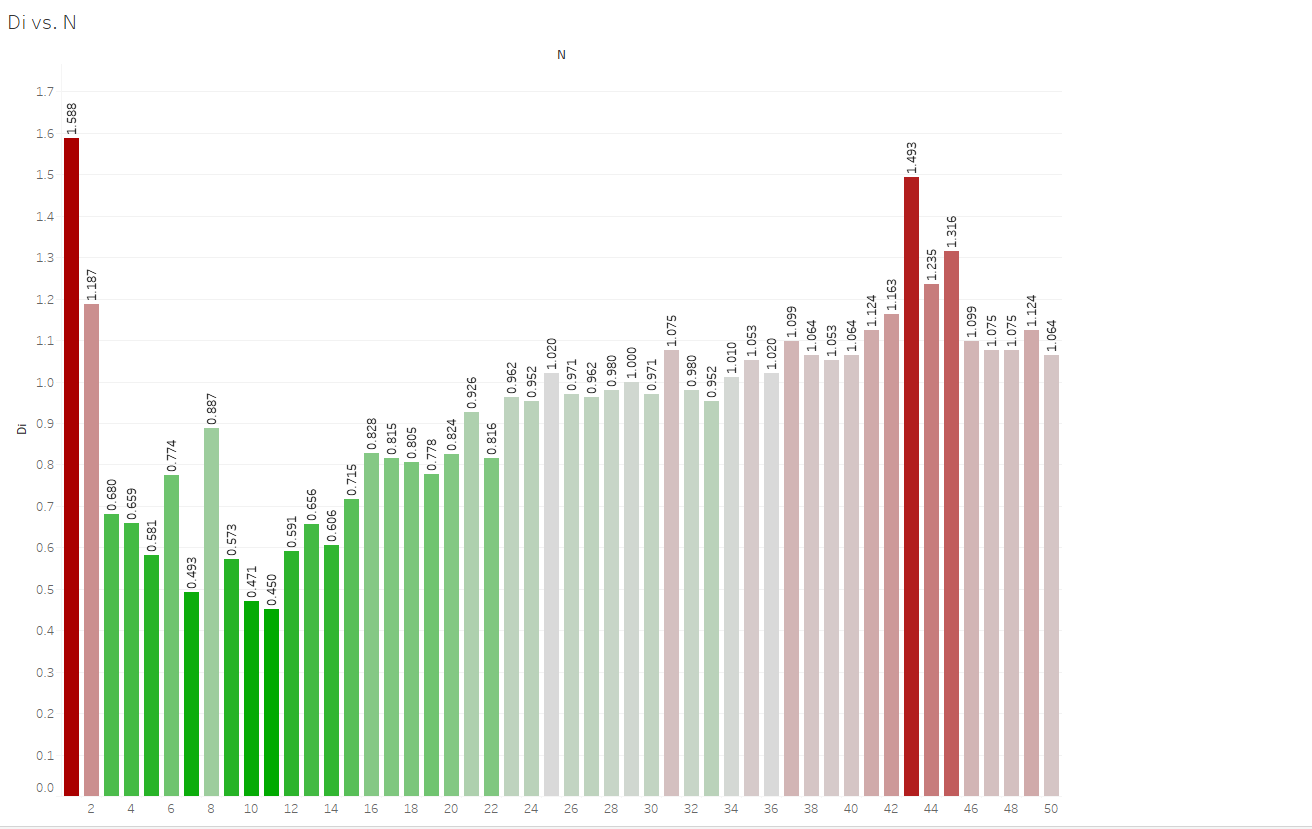
**Di vs. N**

I went about calculating this was via the following method:



This formula for Di is used to calculate the service demands via resource utilizations and system throughput. We are able to retrieve the total time in which a resource is busy by multiplying Ui by T. We can then divide this by the total number of completed requests and get the actual time each resource was busy serving each request.

This yielded the following results:



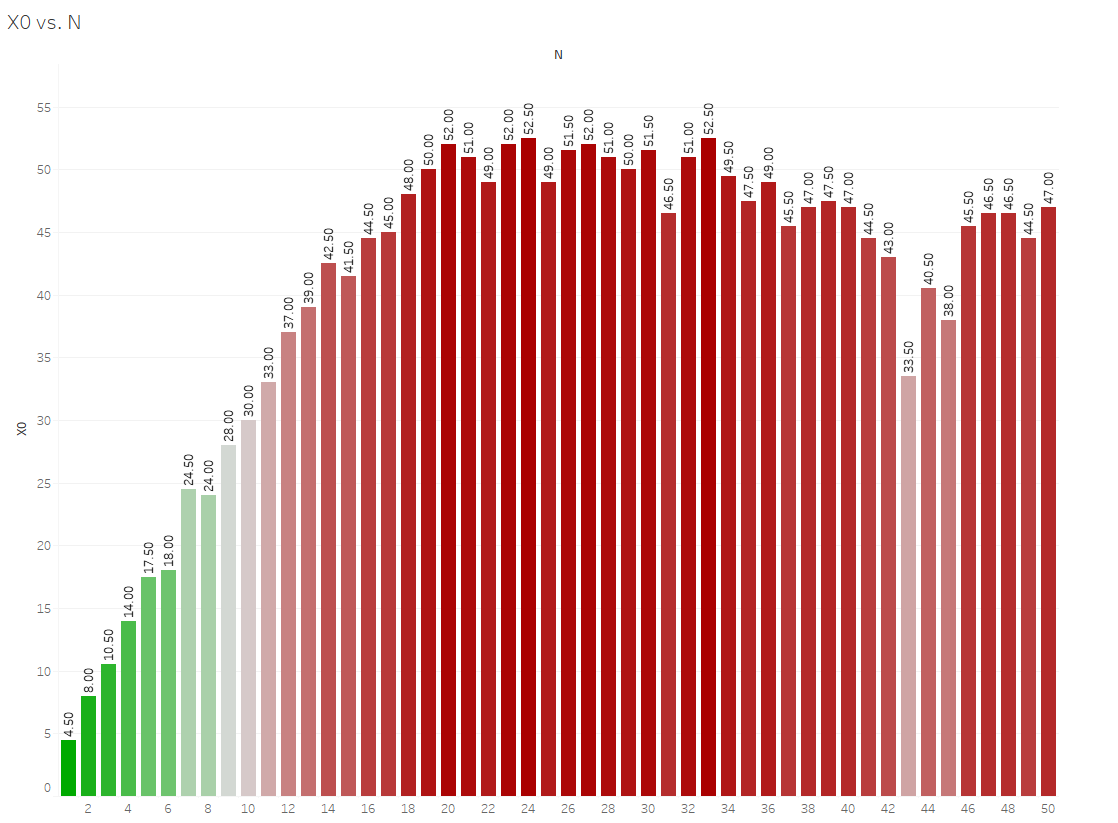
N=1 was the most busy, I guess this was because of the overhead of initially running the tests as this declines over the next few N’s before rising again. Because my script timed out at 2 seconds, none of the values go to 2 or over. The values become higher again towards the end as more resources are being utilized.

**X0 vs. N**

The formula I used for this was:



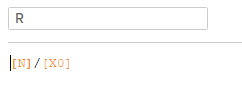
X0 is used to calcuate throughput. In this case we are attempting to calculate the throughput for N.



As we can see from the graph, it is clearly taking a lot longer the bigger N gets. This is due to the higher amount of throughput being produced the longer the program is run.

**R vs. N**

The formula I used was R=N/X0



This gave me the following results:

