report

Tao Wang

2023-01-09

Data cleaning

Firstly we need to do data cleaning for the three datasets, checking the missing values for every column. This table is the summary of missing value in application checkpoints dataset.There is no missing value in the dataset.

Table 1: Missing value in application checkpoints dataset

timestamp	hostname	eventName	eventType	jobId	taskId
0	0	0	0	0	0

This table is the summary of missing value in gpu dataset.There is no missing value in the dataset.

Table 2: Missing value in gpu dataset

timestamp	hostname	gpuSerial	gpuUUID	powerDrawWatt	gpuTempC	gpuUtilPerc	gpuMemUtilPerc
0	0	0	0	0	0	0	0

This table is the summary of missing value in task dataset.There is no missing value in the dataset.

Table 3: Missing value in task dataset

taskId	jobId	x	y	level
0	0	0	0	0

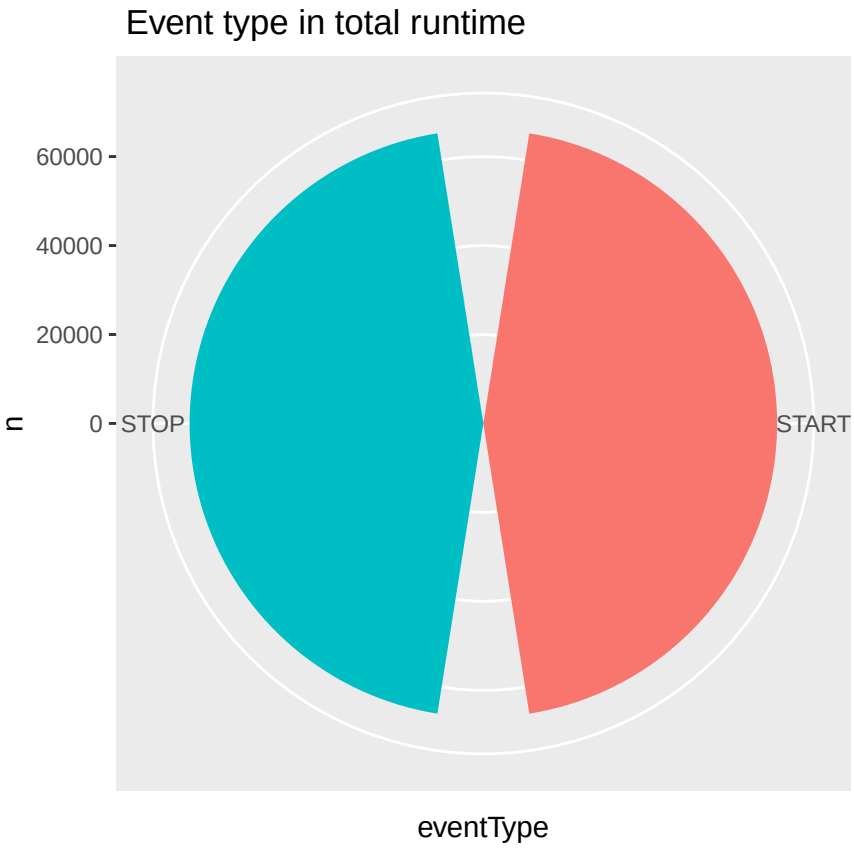
Which event types dominate task runtimes?

The table below shows sum of event type in total runtime,the number of starts and stops is equal throughout the total render process.

Table 4: Event type in total runtime

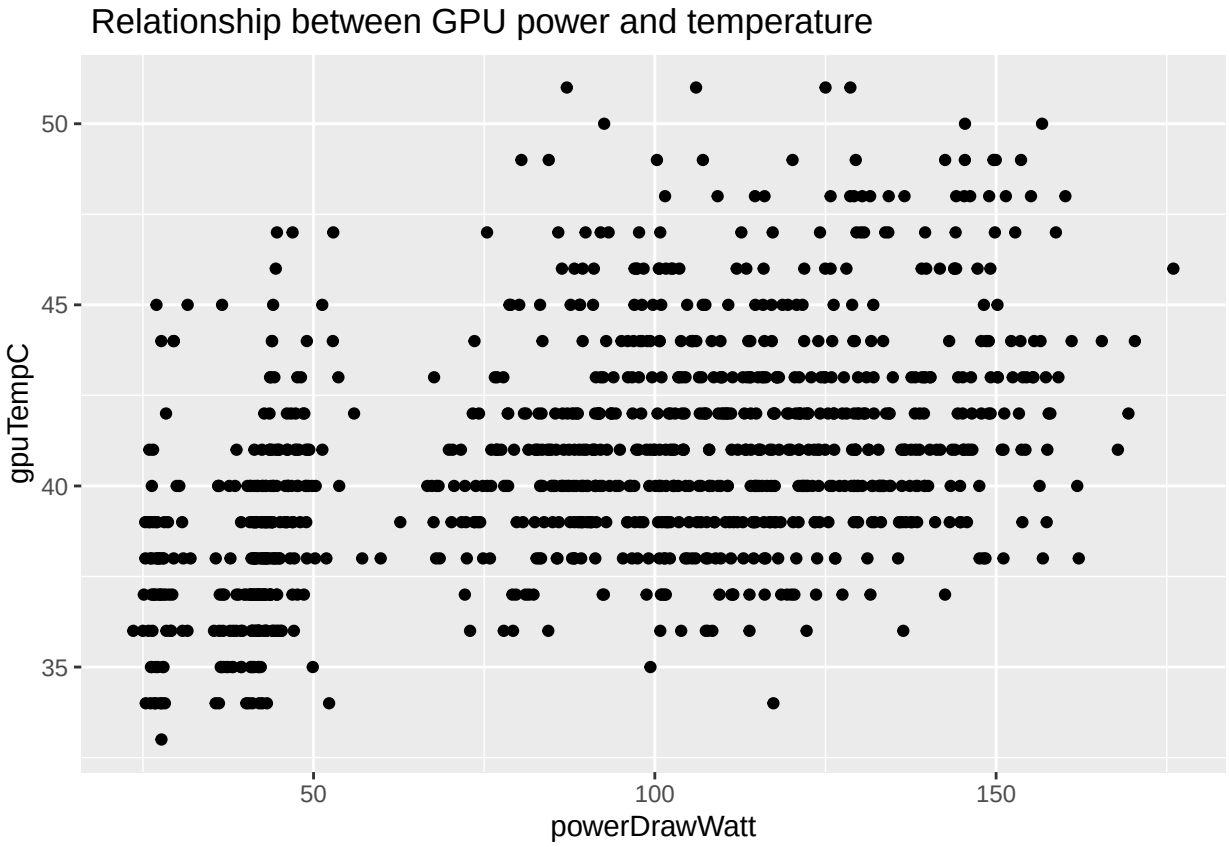
Event type	Sum
START	66040
STOP	66040

The pie chart shows sum of event type in total runtime.



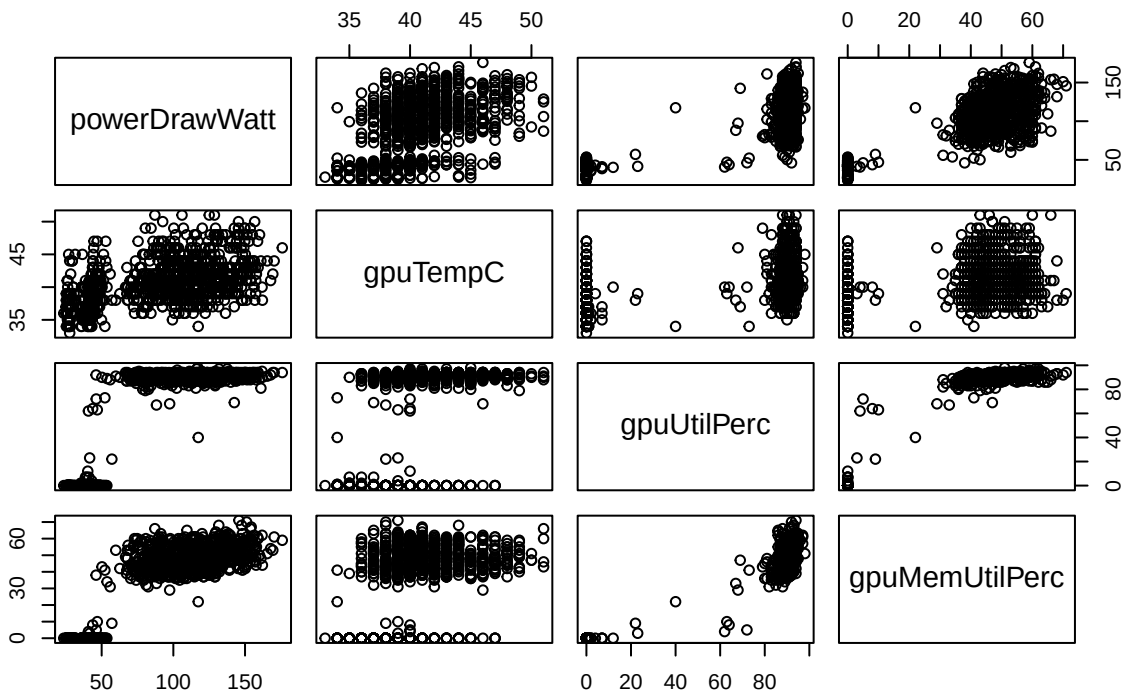
What is the interplay between GPU temperature and performance?

The following scatter plot shows the relationship between GPU power and temperature.There is no obvious linear relationship between GPU power and temperature.



The scatterplot matrix shows the relationship among GPU performance variables.There is no obvious linear relationship among these variables.But for power and percent utilisation of GPU memory, when percent utilisation of GPU memory is above 30%, there is a slight linear relationship between it and GPU power.

Relationship among GPU performance variables



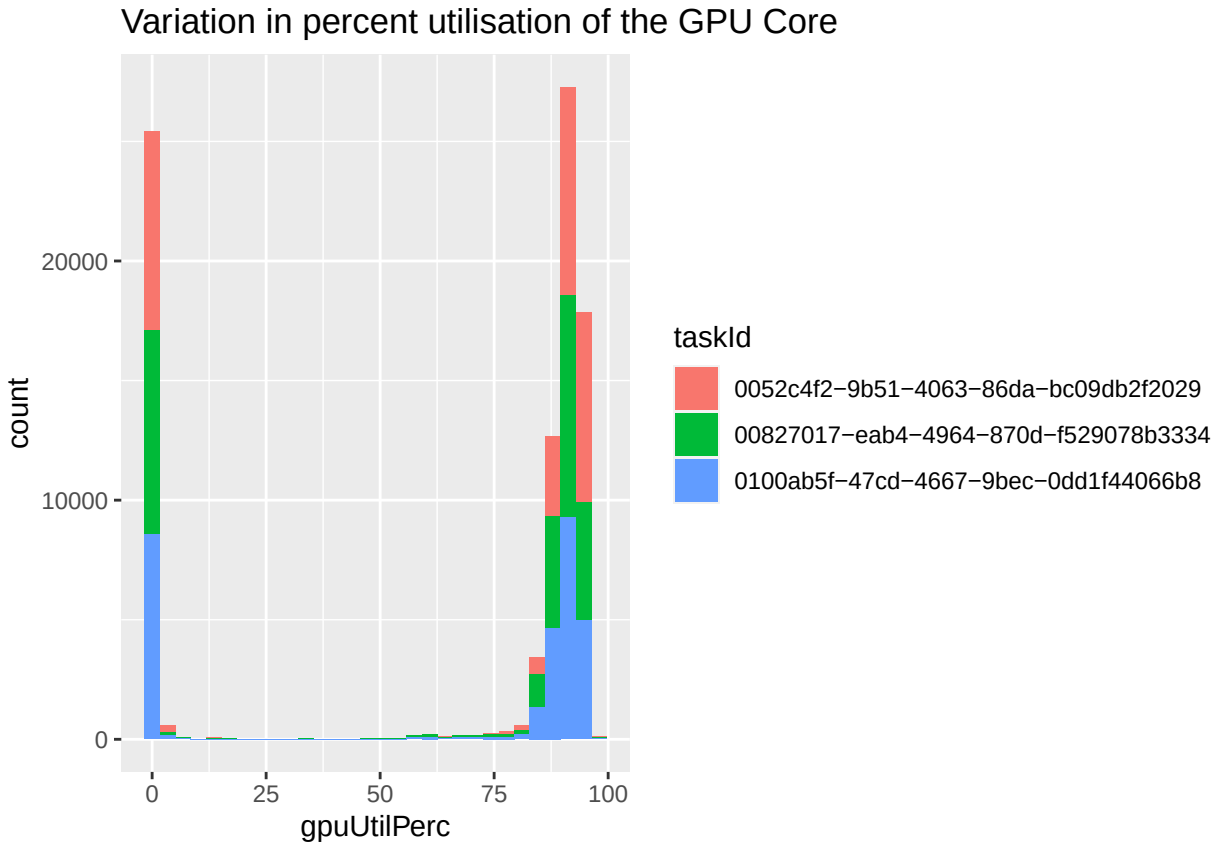
Can we quantify the variation in computation requirements for particular tiles?

The following table shows the TOP 10 sum of tiles, we randomly select three tiles as representatives.

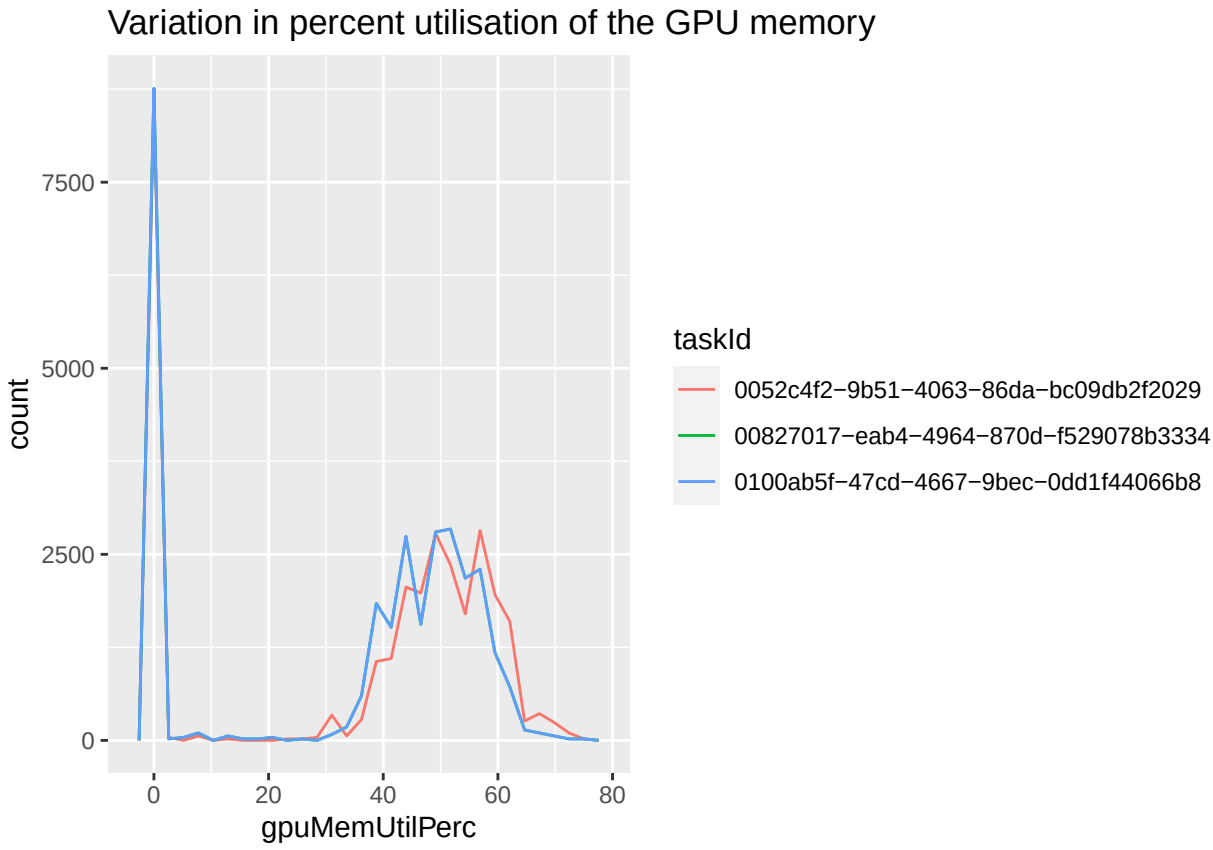
Table 5: TOP 10 sum of tiles

Tiles	Sum
0052c4f2-9b51-4063-86da-bc09db2f2029	20
00827017-eab4-4964-870d-f529078b3334	20
0100ab5f-47cd-4667-9bec-0dd1f44066b8	20
01a66eb8-6ff2-45c0-a5d8-aea4a26acf46	20
0513521f-1446-4013-bfc3-ba102b673015	20
054a491f-1317-43ac-a80a-1eb831442dbc	20
05aa742f-4089-4c07-8eee-4971cf0f7d40	20
06c55088-5cc0-42c6-b891-86203a274005	20
06f82c18-1771-4576-a0d3-5d5f39f98860	20
06ff2ffa-c05d-4b37-bbd2-4a8a0a578ce0	20

For these three tiles, the histogram below shows the variation in percent utilisation of the GPU Core, data is mainly distributed at 0% and after 80%.For different tiles, percent utilisation of the GPU Core vary greatly.When percent utilisation of the GPU Core is near 85%, red tile's data is almost three times as big as blue tile's data.



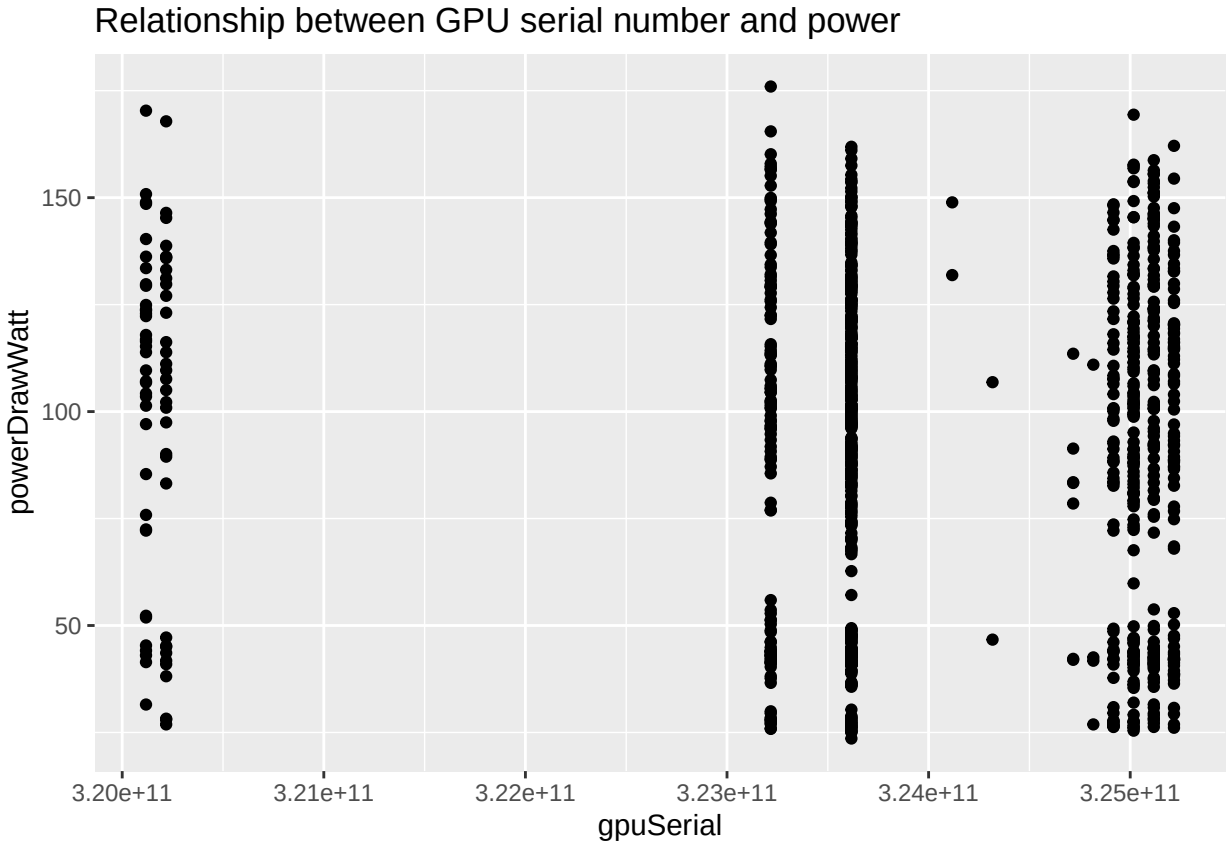
For these three tiles,the line chart below shows the variation in percent utilisation of the GPU memory, data is mainly distributed at 0% and between 30% and 70%.For different tiles, there is no obvious difference in percent utilisation of the GPU memory.



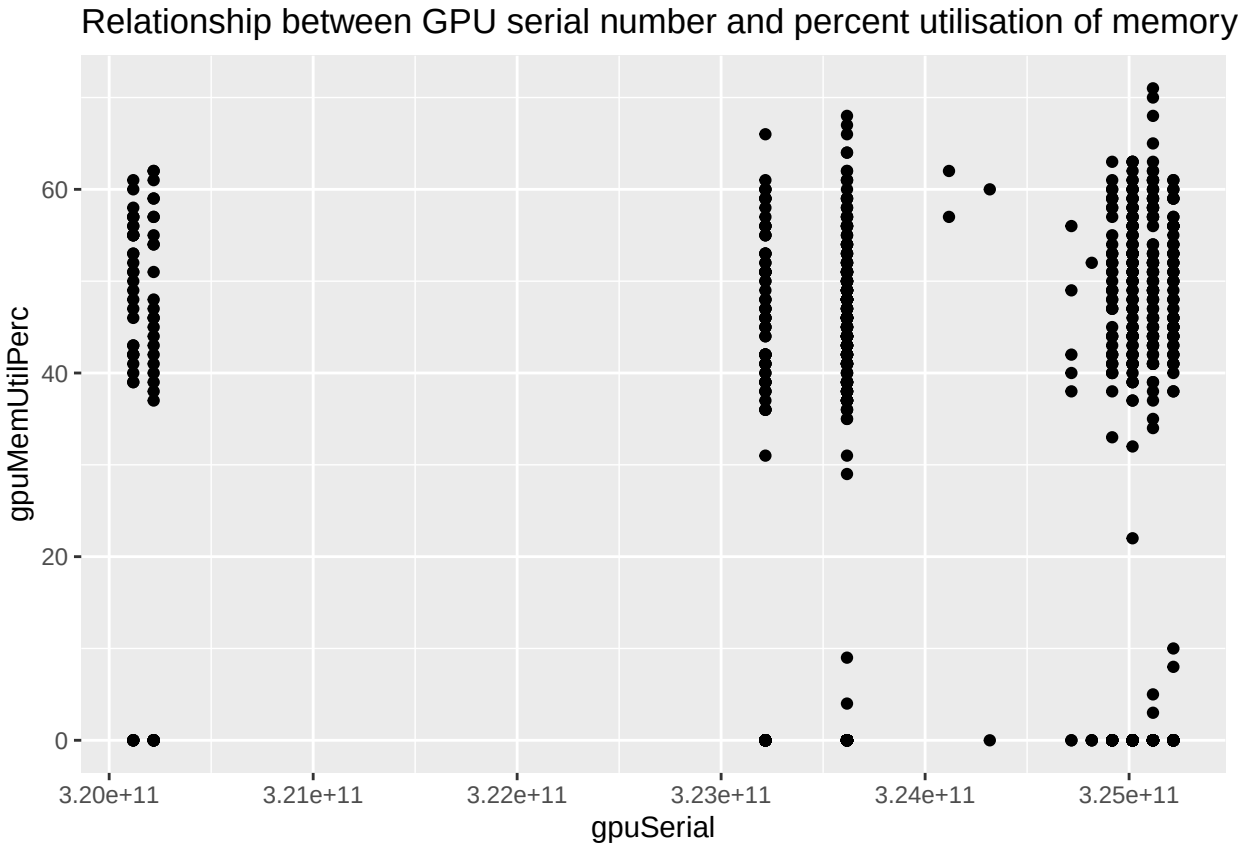
Can we identify particular GPU cards (based on their serial numbers) whose performance differs to other cards

Now let's explore the relationship between GPU serial number and performance.

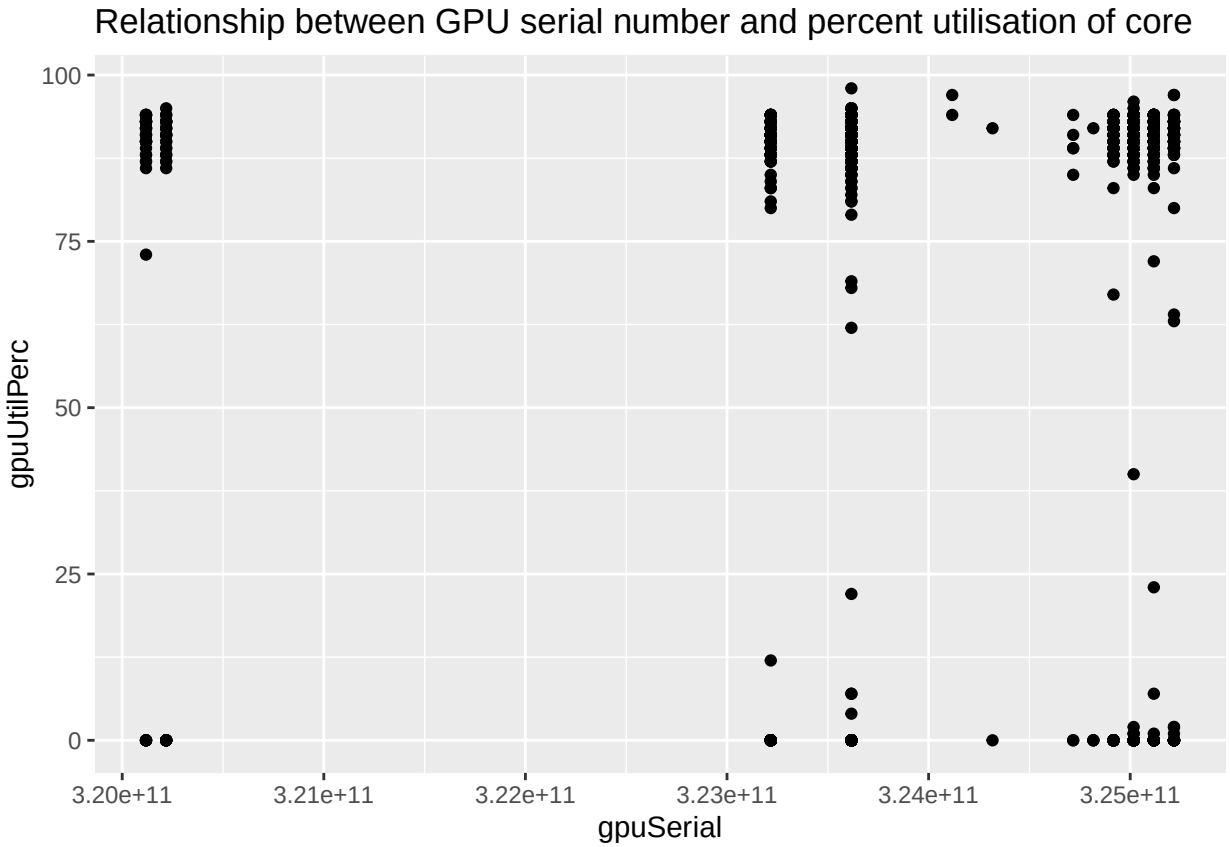
The following scatter plot shows the relationship between GPU serial number and power.There is no obvious relationship between them.



The following scatter plot shows the relationship between GPU serial number and percent utilisation of memory.There is no obvious relationship between them.



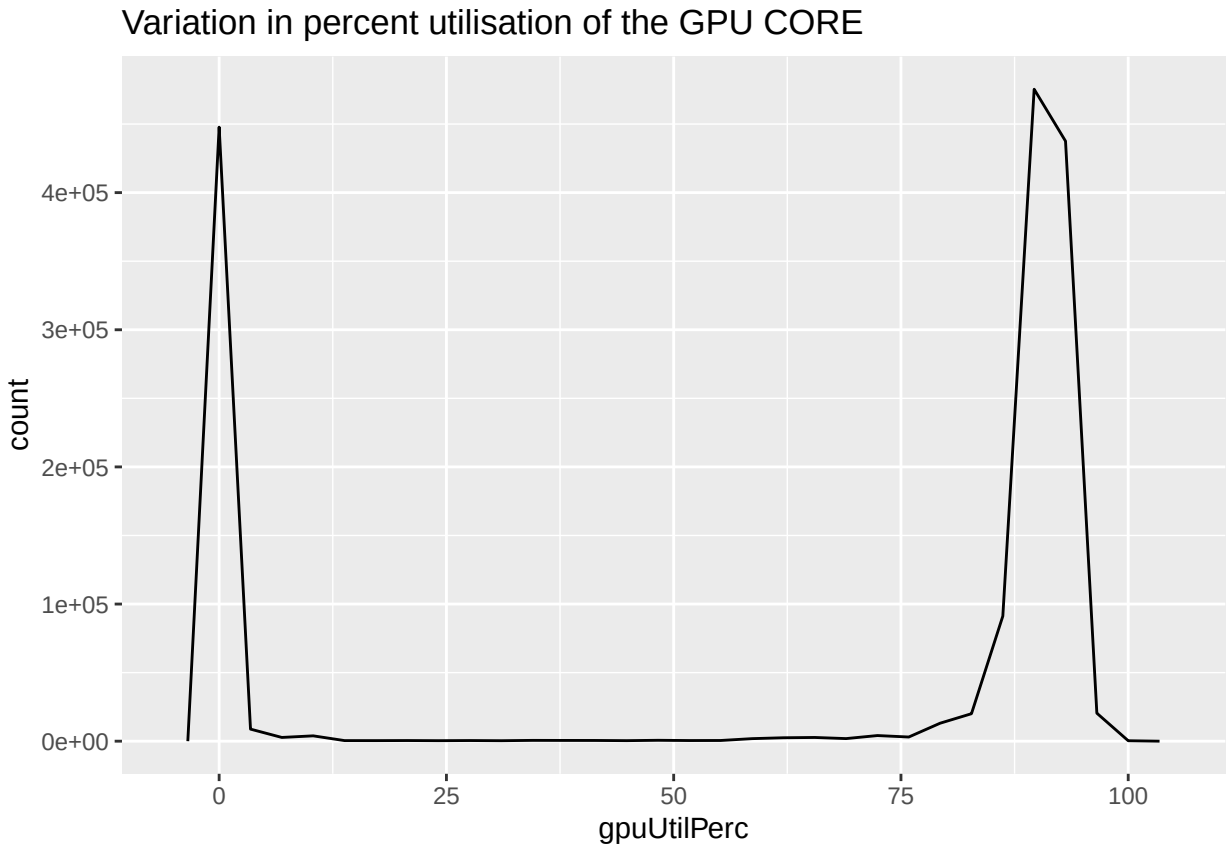
The following scatter plot shows the relationship between GPU serial number and percent utilisation of core.There is no obvious relationship between them.


Relationship between GPU serial number and percent utilisation of core
gpuUtilPerc
100
75
50
25
0
3.20e+11
3.21e+11
3.22e+11
3.23e+11
3.24e+11
3.25e+11
gpuSerial

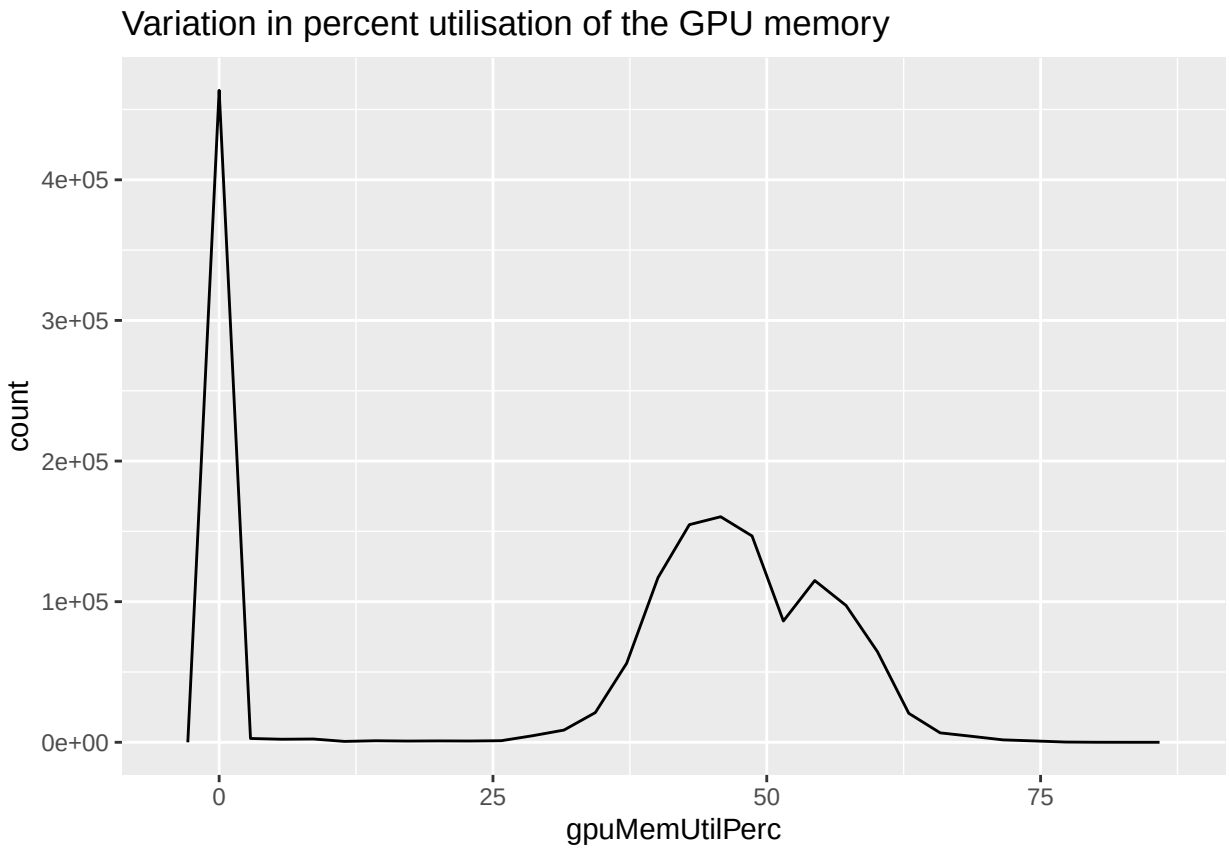
What can we learn about the efficiency of the task scheduling process

When it comes to the efficiency of the task scheduling process,we mainly explore variation in percent utilisation of the GPU Core and memory for all GPU cards.

For all GPU cards,the line chart below shows the variation in percent utilisation of the GPU CORE, the data is mainly distributed at 0% and between 75% and 100%.



For all GPU cards,the line chart below shows the variation in percent utilisation of the GPU memory, the data is mainly distributed at 0% and between 25% and 70%.

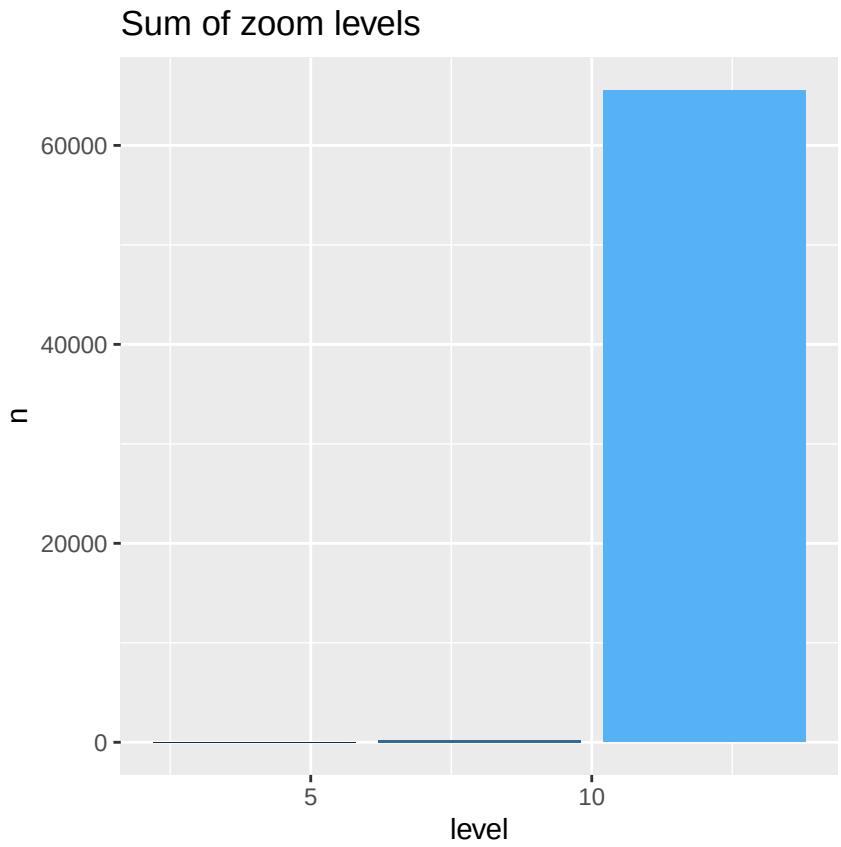


The table below shows the sum of zoom levels. Level 12 accounts for the majority of the data set which is 65536, while level 8 and 4 only have 256 and 1 respectively.

Table 6: Sum of zoom levels

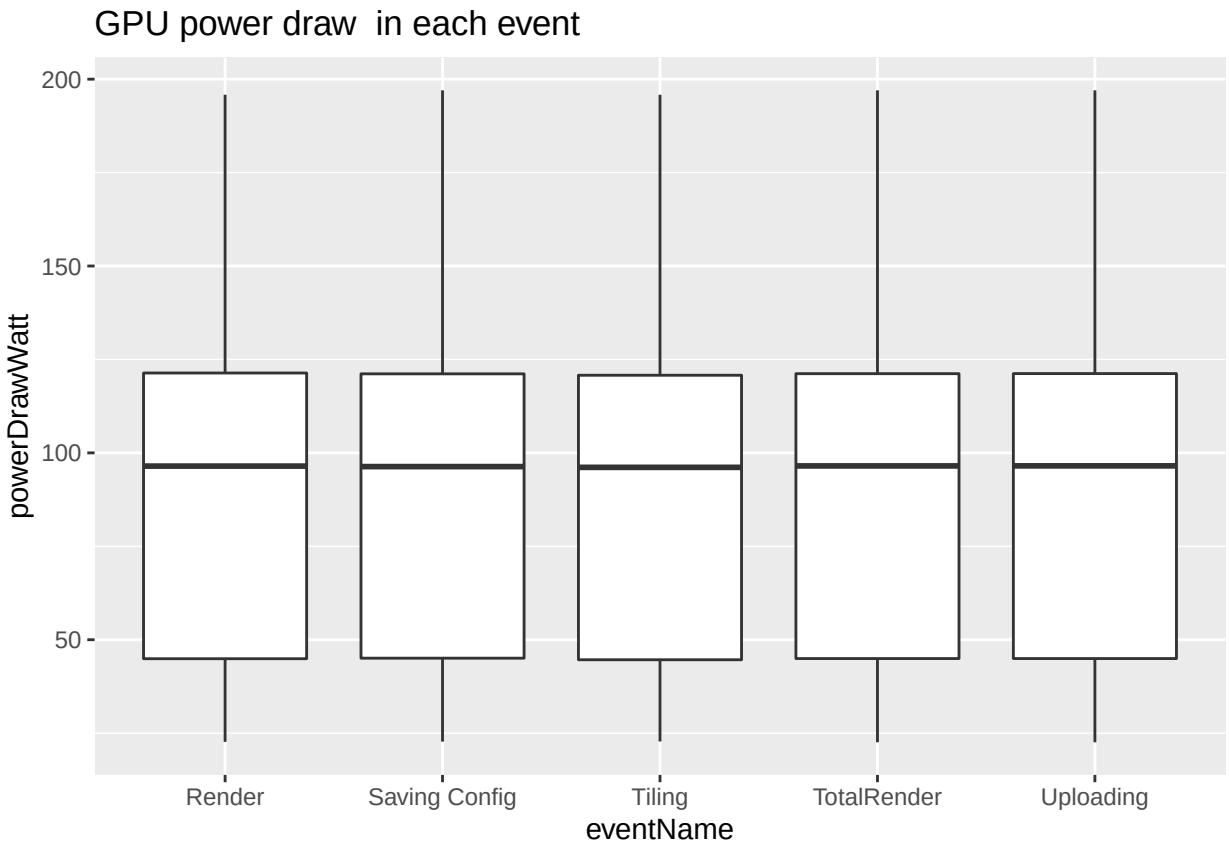
Zoom level	Sum
12	65536
8	256
4	1

The bar chart below shows the sum of zoom levels



Now let's look at the distribution of GPU power draw in each event within the rendering application.

The boxplot below shows GPU power draw in five different events.Power distribution is basically consistent.First quartile is about 50 watt,Median is about 100 Watt,Third quartile is about 125 Watt.

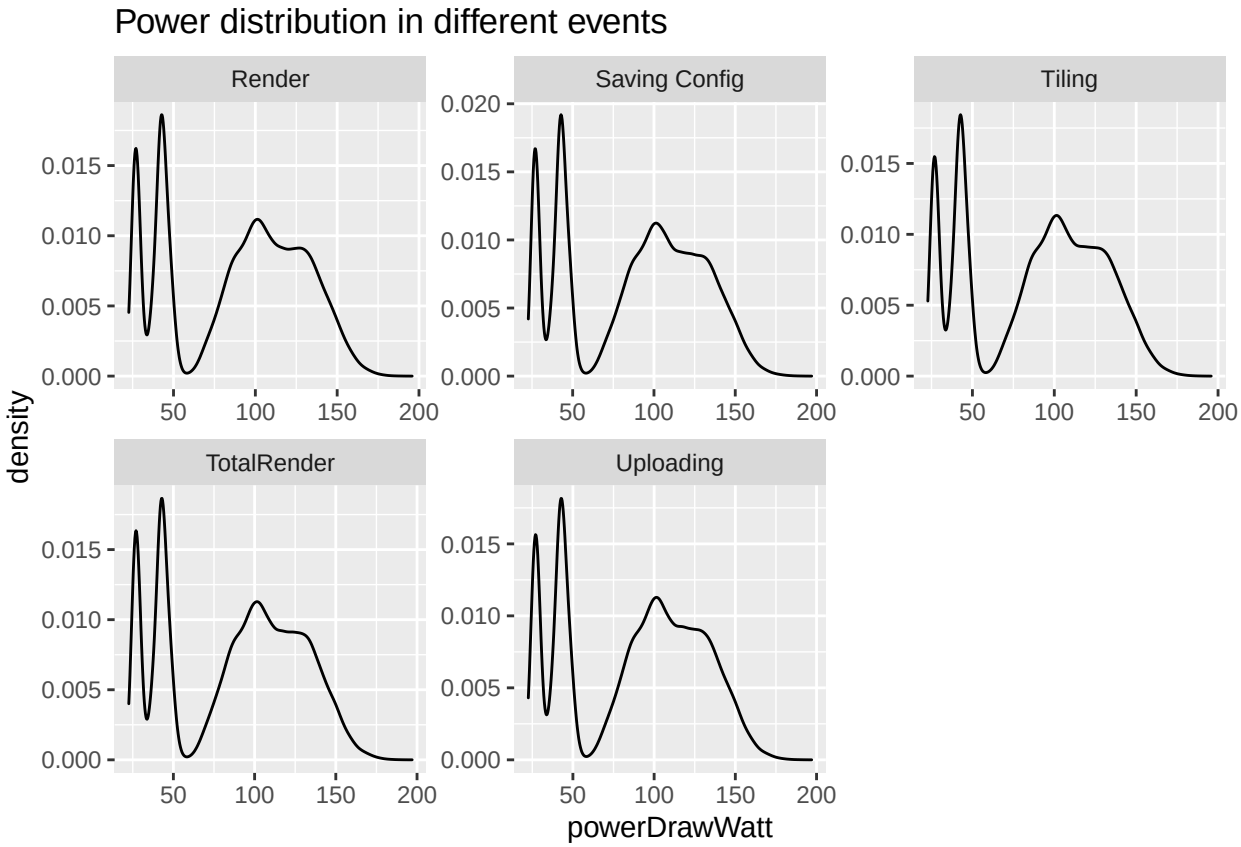


The table below shows average performance parameters in five different events.For these four parameters, the average performance data is almost equal.

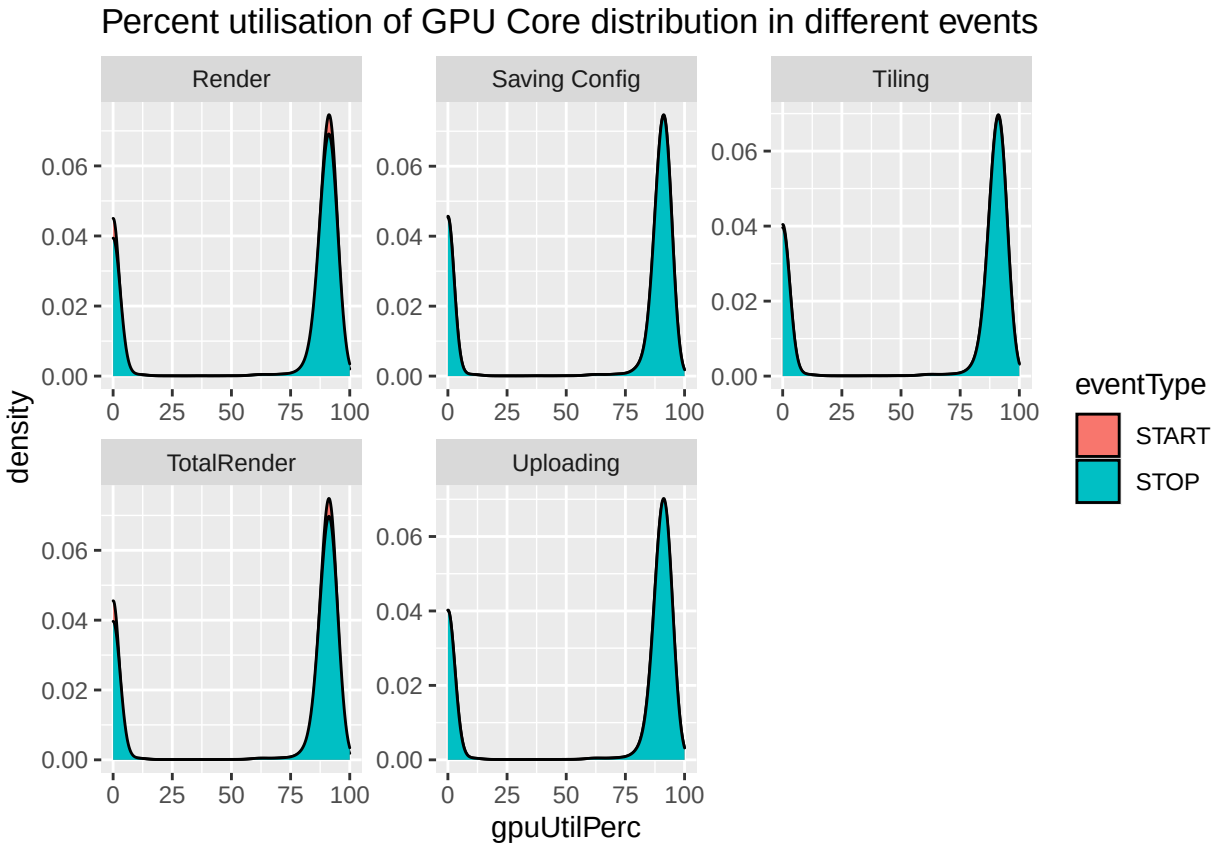
Table 7: Event type in total runtime

eventName	powerDrawWatt	gpuTempC	gpuUtilPerc	gpuMemUtilPerc
Render	89.16400	40.07951	63.03159	33.34738
Saving Config	89.10278	40.06120	62.86650	33.06840
Tiling	88.81525	39.73460	62.92436	33.18058
TotalRender	89.14005	40.03969	63.00752	33.29650
Uploading	89.15840	39.91538	62.98529	33.27147

The density diagram below shows the power distribution in different events.Power distribution is basically consistent,in each event, the most commonly used power is below 50.



The density diagram below shows the percent utilisation of GPU Core distribution in different events.Percent utilisation of GPU Core distribution is basically consistent,in each event, the most common percent utilisation is between 75% and 100%. For render and total render event, Percent utilisation of start type is slightly higher than stop type.


Percent utilisation of GPU Core distribution in different events
Render
Saving Config
Tiling
TotalRender
Uploading
density
gpuUtilPerc
0.00
0.02
0.04
0.06
0
25
50
75
100
eventType
START
STOP