# Graphite Quota/Usage: metrics of metrics

This is mostly about an reability improvement in the Graphite storage running at Bookign.com: Go-Carbon.

## What is Graphite and Graphite at Booking.com

Graphite is a timeseries database.

- 400 millions uniq metrics (stale metrics are being constantly pruned)
- 700 TB of disk usage
- 1000+ query requests per second

### **WHY**

To keep Graphite the system alive where there are "unscheduled" metric growth or bugs that produces high amoutn of metrics.

## **HOW**

Maintaining the usage on the index trie tree

(Concurrent/Realtime Trie Index + Ingestion Dropping)

## What's Graphite Quota/Usage

```
[general]
metrics = 1,000,000
throughput = 60,000,000 # 50GB

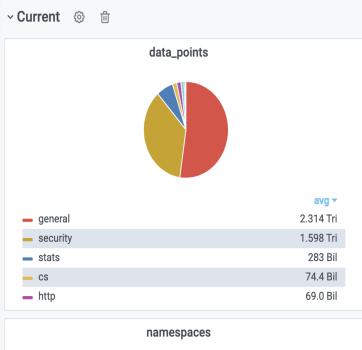
[/]
metrics = 10,000,000
throughput = 600,000,000
physical_size = 250,000,000,000 # 250GB
```

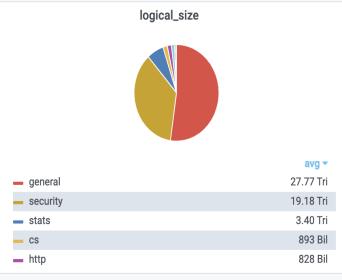
### **WHERE**

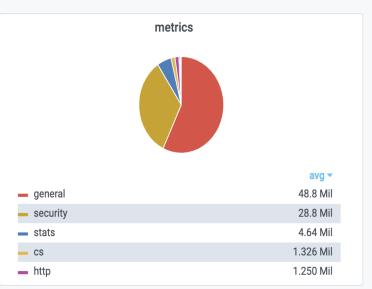
For now, we are just looking at enabling it on root level.

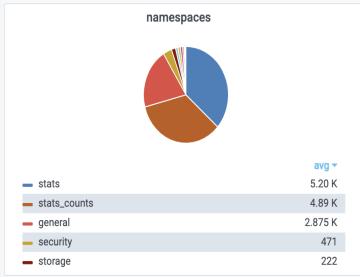
But the usage of various namespaces/prefixes that are collected to give us visibility of them.

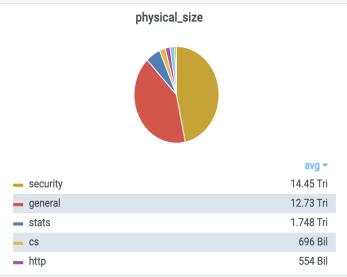
If needed, we can quickly enable throttling on the targeted namespaces.

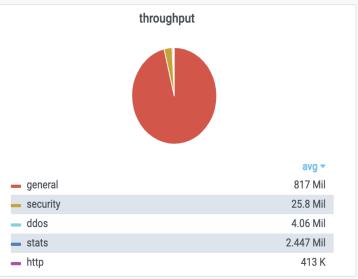




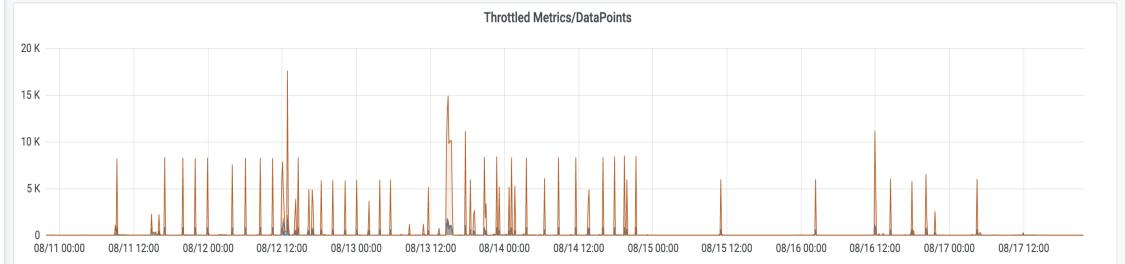








v Quota Usages 🐵 🗓 physical\_size quota usage throughput quota usage metrics quota usage v Details



### Bonus

Go built-in hashmap is much faster than my hand-bake trie tree traversing.

It actually saves us 5 CPU cores when it comes to throughput throttling.

In highsight, this is common wisdom as hashmap has o(1) lookup.