Monitoring with InfluxDB and Grafana

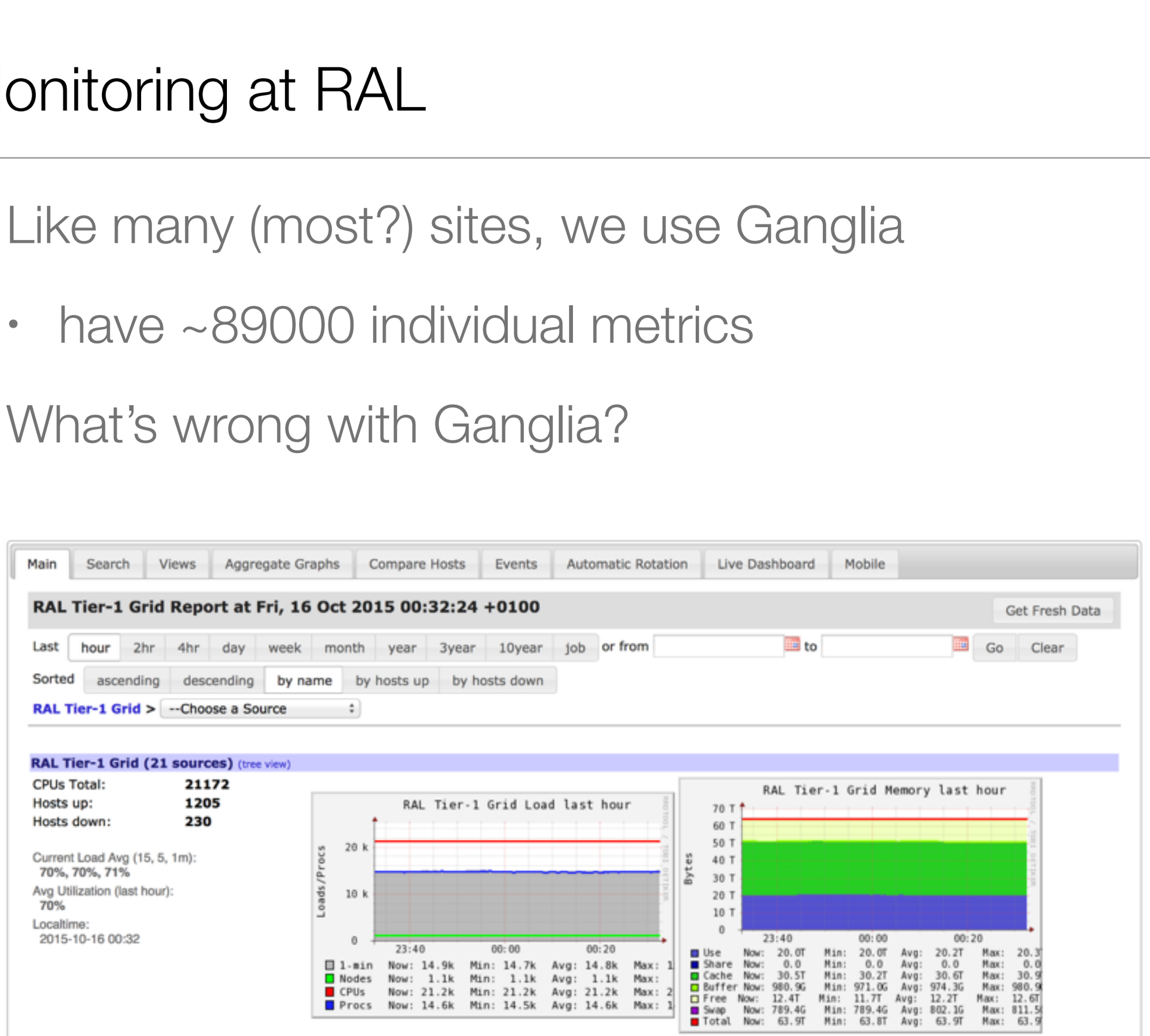
Andrew Lahiff
STFC RAL

HEPiX 2015 Fall Workshop, BNL

Introduction

Monitoring at RAL

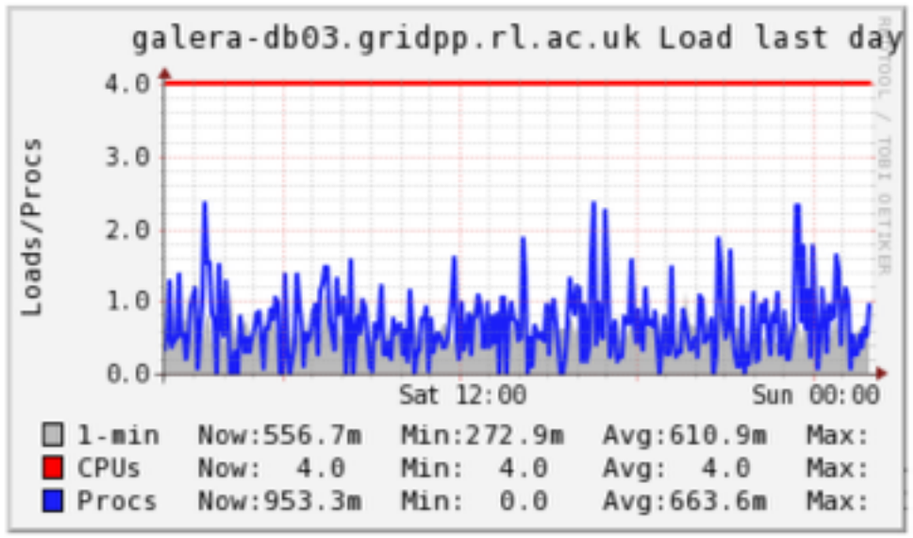
- Like many (most?) sites, we use Ganglia
  - have ~89000 individual metrics
- What's wrong with Ganglia?

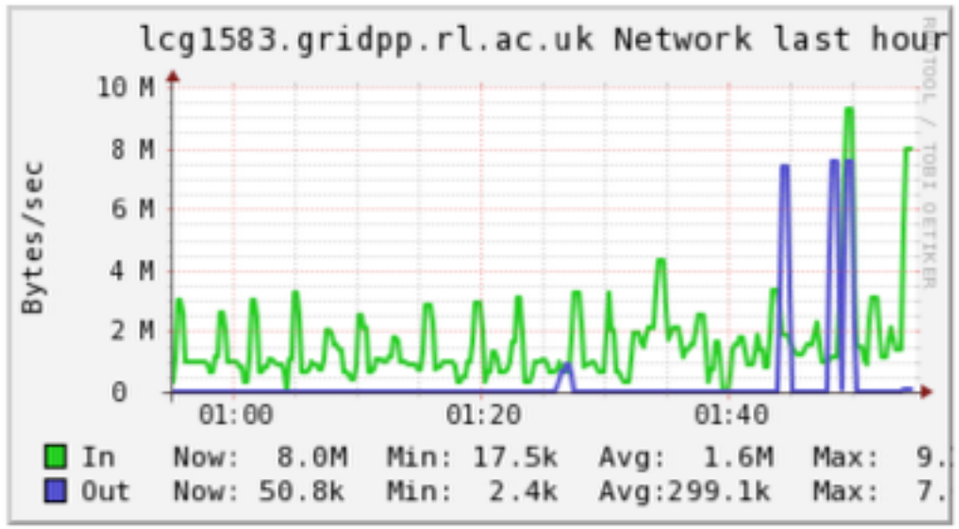


STFC

Problems with ganglia

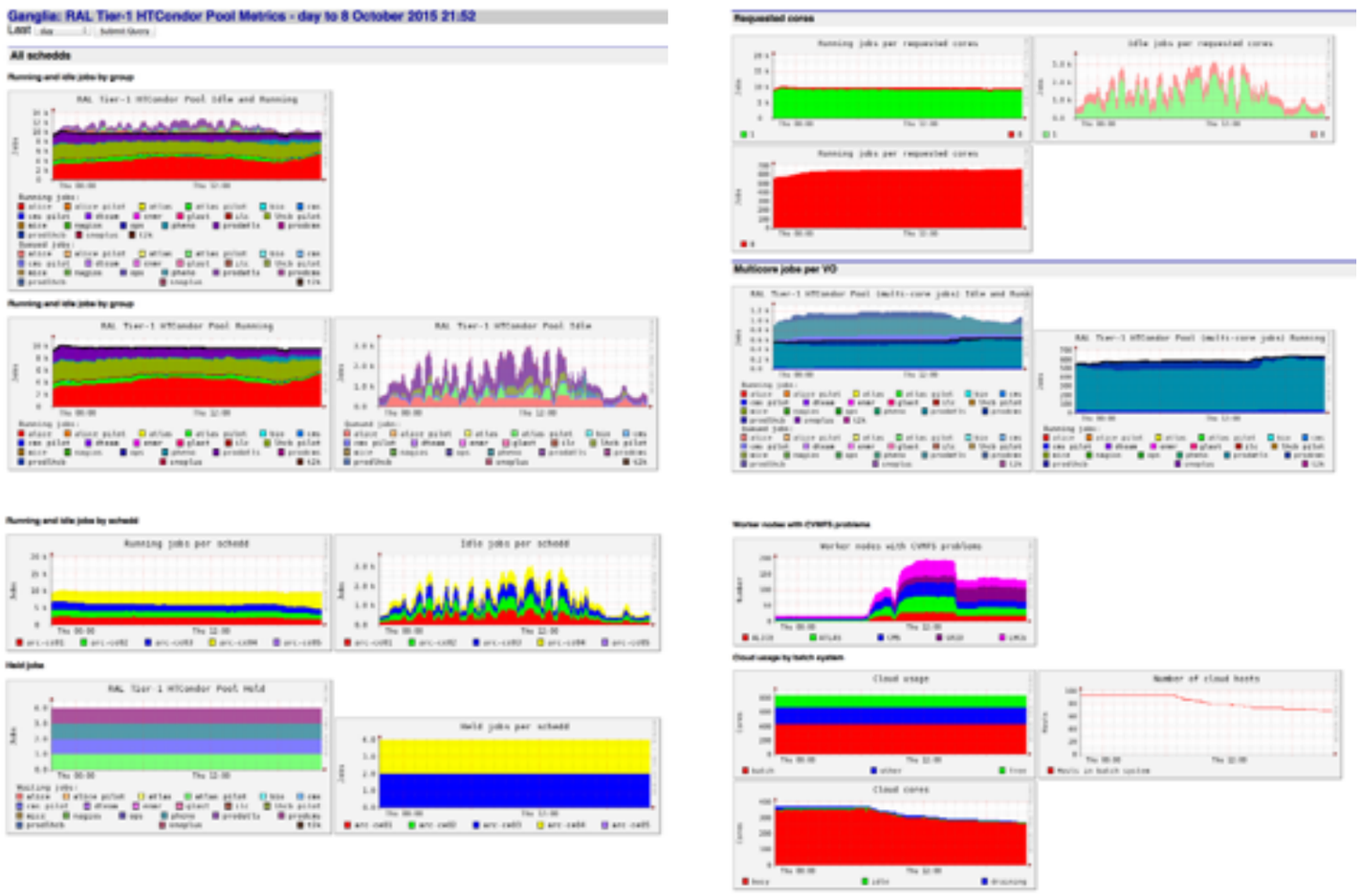
- Plots look very dated





Problems with ganglia

- Difficult & time-consuming to make custom plots
  - currently use long, complex, messy Perl scripts
  - e.g. HTCondor monitoring > 2000 lines



Problems with ganglia

- Difficult & time-consuming to make custom plots
  - Ganglia UI for making customised plots is restricted & doesn't give good results

STFC

Problems with ganglia

- Ganglia server has demanding host requirements
  - e.g. we store all rrds in a RAM disk
  - have problems if trying to use a VM
- Doesn’t handle dynamic resources well
- Occasional problems with gmond using too much memory, affecting other processes on machines
- Not really suitable for Ceph monitoring

A possible alternative

- InfluxDB + Grafana
  - InfluxDB is a time-series database
  - Grafana is a metrics dashboard
    - originally a fork of Kibana
    - can make plots of data from InfluxDB, Graphite, others…
- Very easy to make (nice) plots
- Easy to install

STFC

InfluxDB

- Time series database written in Go
- No external dependencies
- SQL-like query language
- Distributed
  - can be run as a single node
  - can be run as a cluster for redundancy & performance (not suitable for production use yet)
- Data can be written in using REST, or an API (e.g. Python)
  - or from collectd or graphite

STFC

InfluxDB

- Data organised by time series, grouped together into databases
- Time series have zero to many points
- Each point consists of:
  - **time** - the timestamp
  - a **measurement** (e.g. cpu_load)
  - at least one key-value **field**, e.g. value=0.15 or 5min=0.78
  - zero to many **tags**, containing metadata, e.g. host=lcg1451

STFC

InfluxDB

- Points written into InfluxDB using line protocol format

```
<measurement>[,<tag-key>=<tag-value>...] <field-key>=<field-value>[,<field2-key>=<field2-value>...] [timestamp]
```

- Example for an FTS3 server

```
active_transfers,host=lcgfts01,instance=production,vo=atlas value=21
```

- Can write multiple points in batches to get better performance, e.g. 2000 points (0.9.4):
  - sequentially: **245s**
  - batch: **0.357s**

STFC

Examples

HTCondor

- Metrics from condor_gangliad; in HTCondor config

```
GANGLIA_GMETRIC = /usr/local/bin/htcondor2influx.pl
```

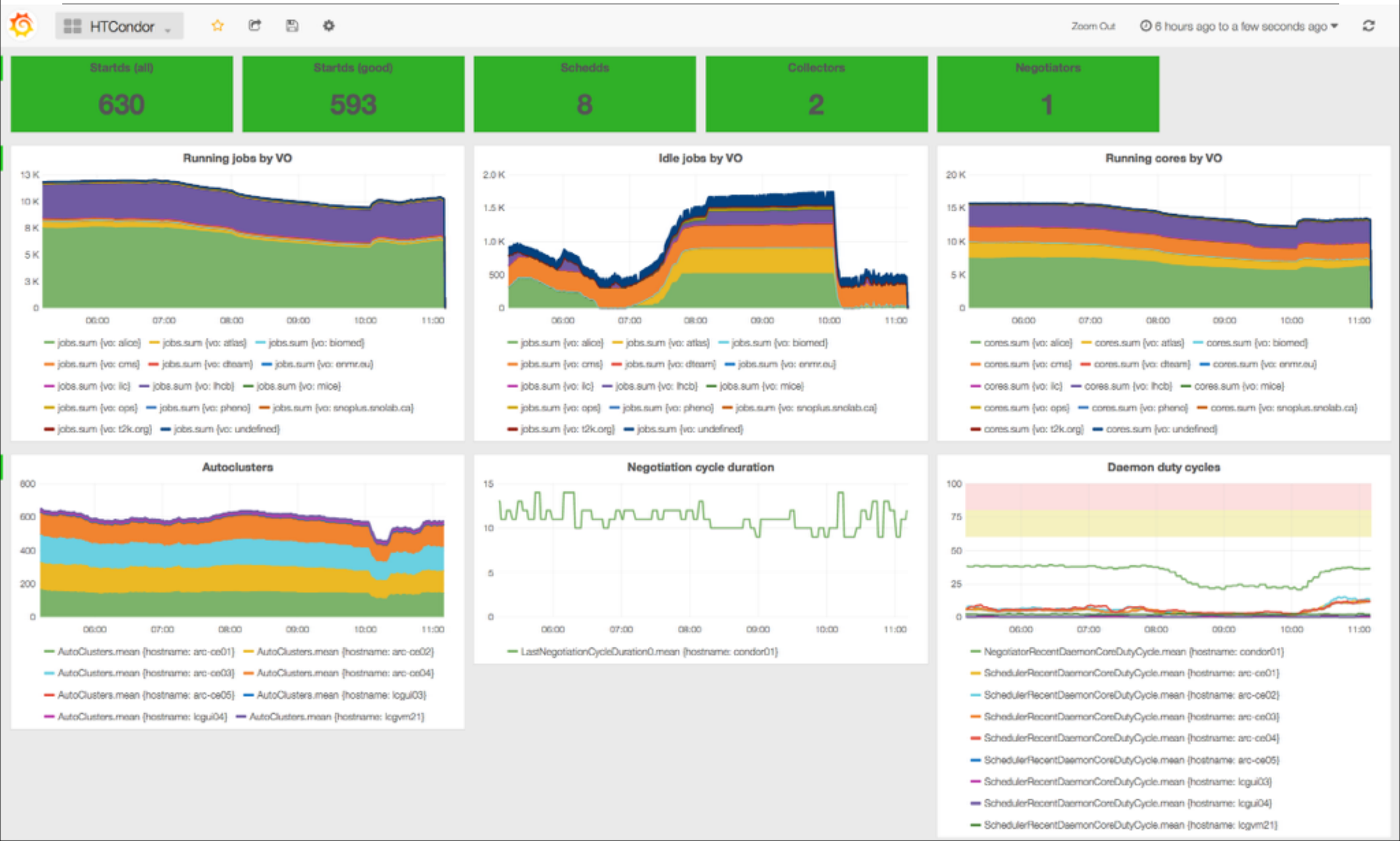
  - Problem: sends metrics individually (not in batches)

- Also custom metrics via cron + Python script

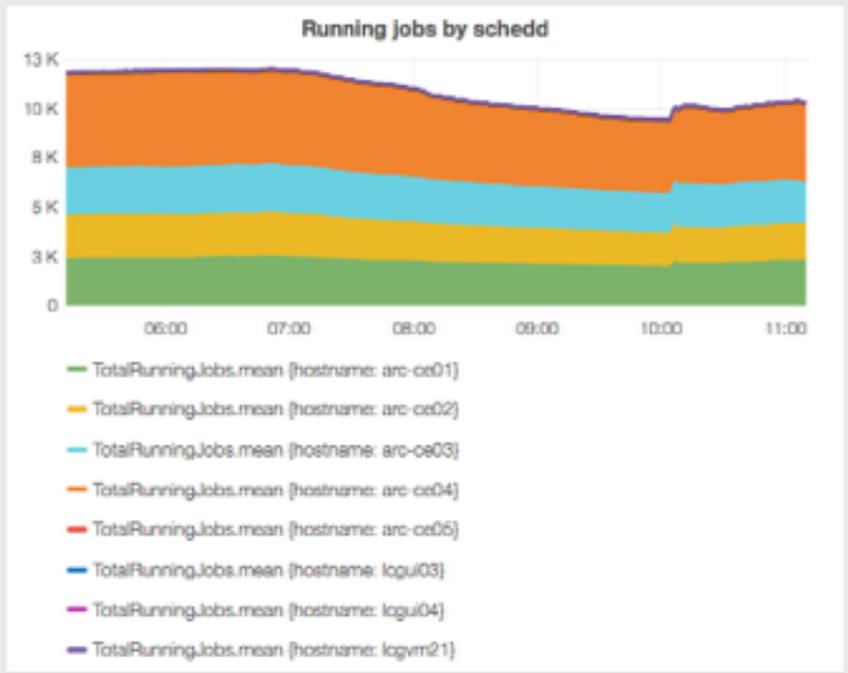
  - e.g. jobs by VO

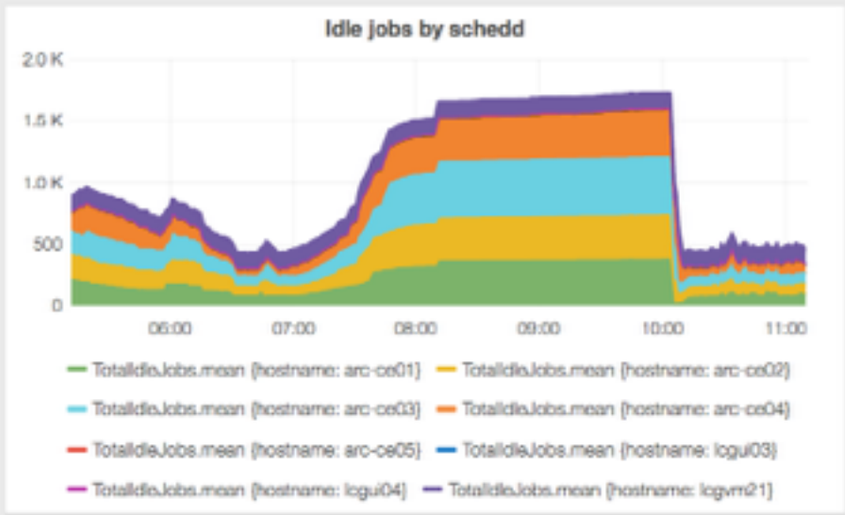
STFC

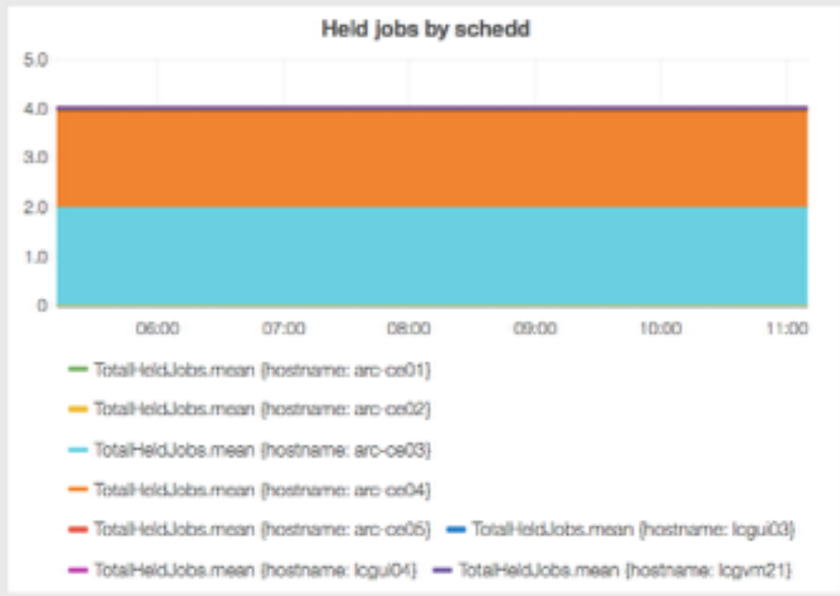
(Preliminary) HTCondor dashboard

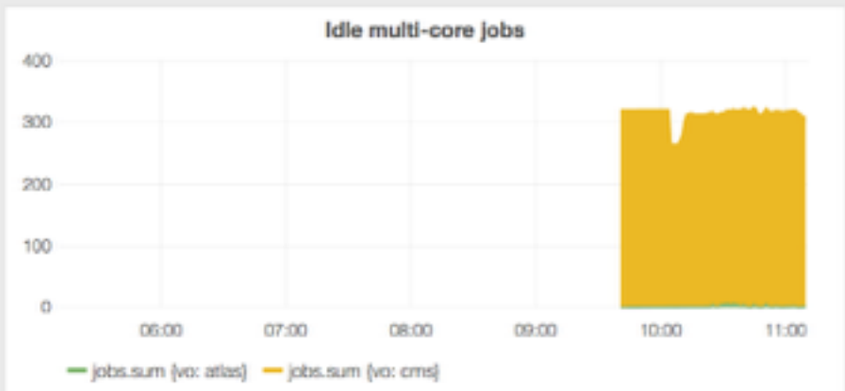


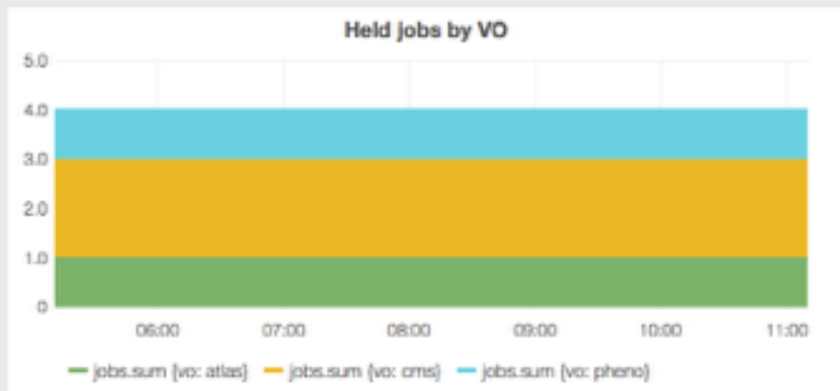
(Preliminary) HTCondor dashboard

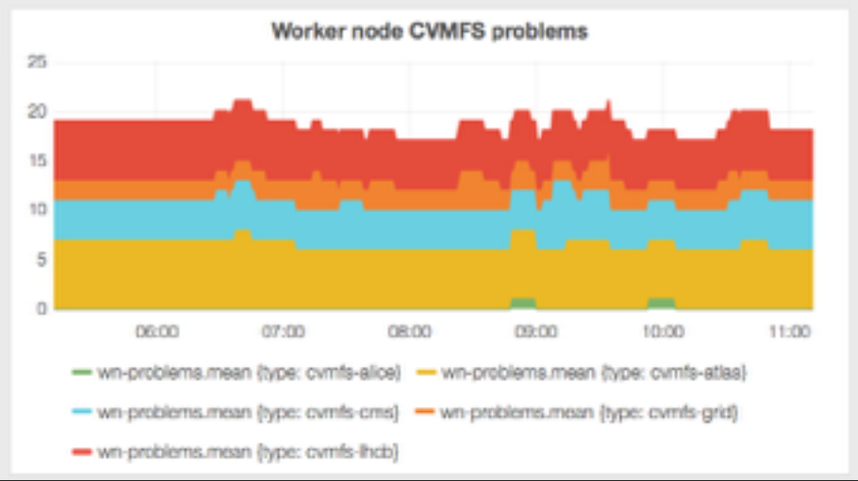


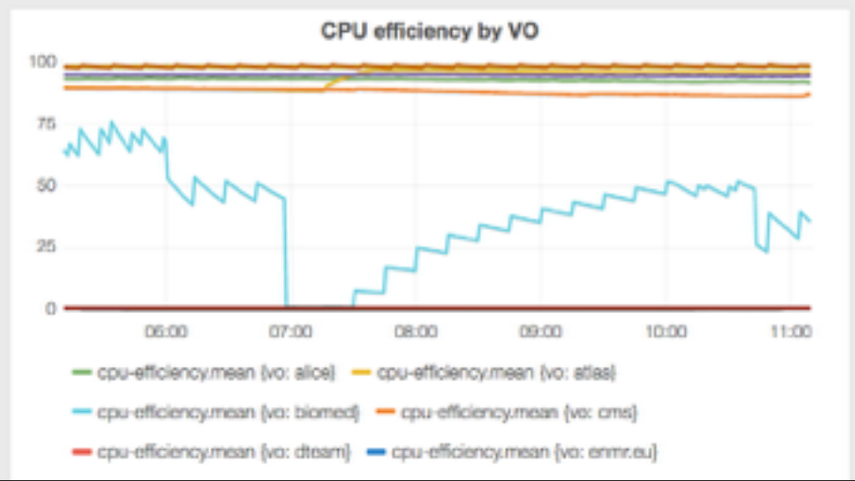






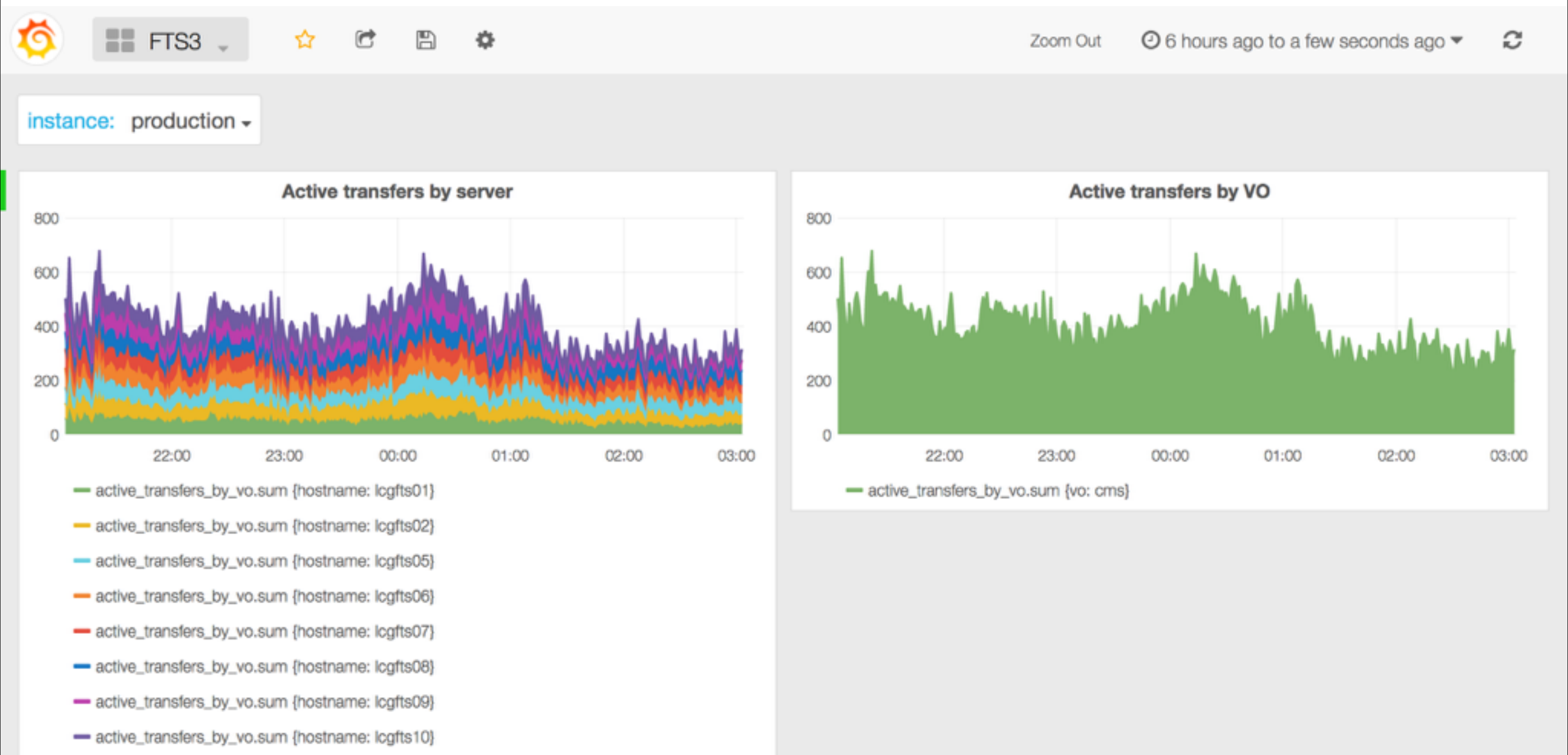






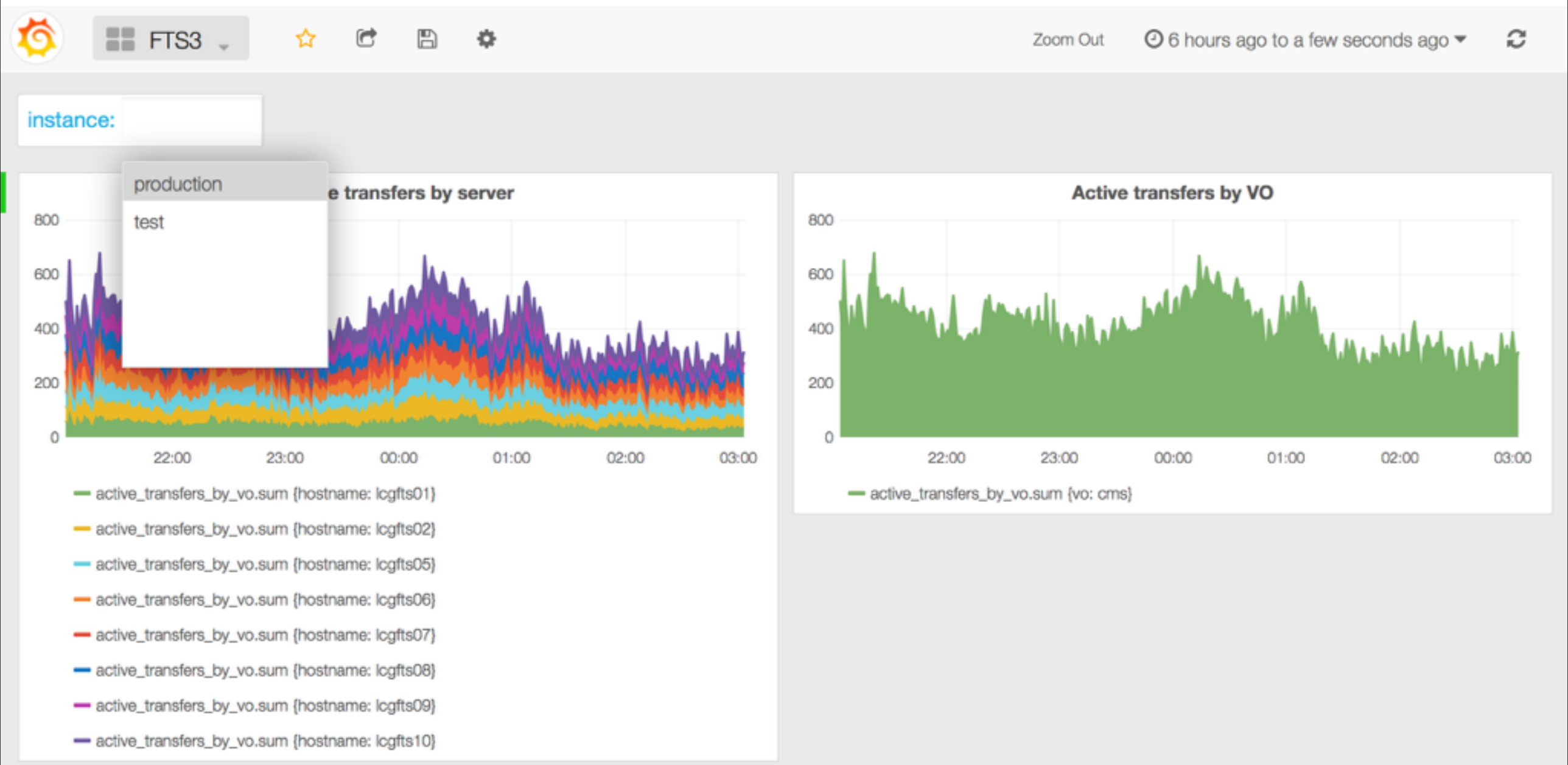
Grafana templating example - FTS3

- View of “production” instance



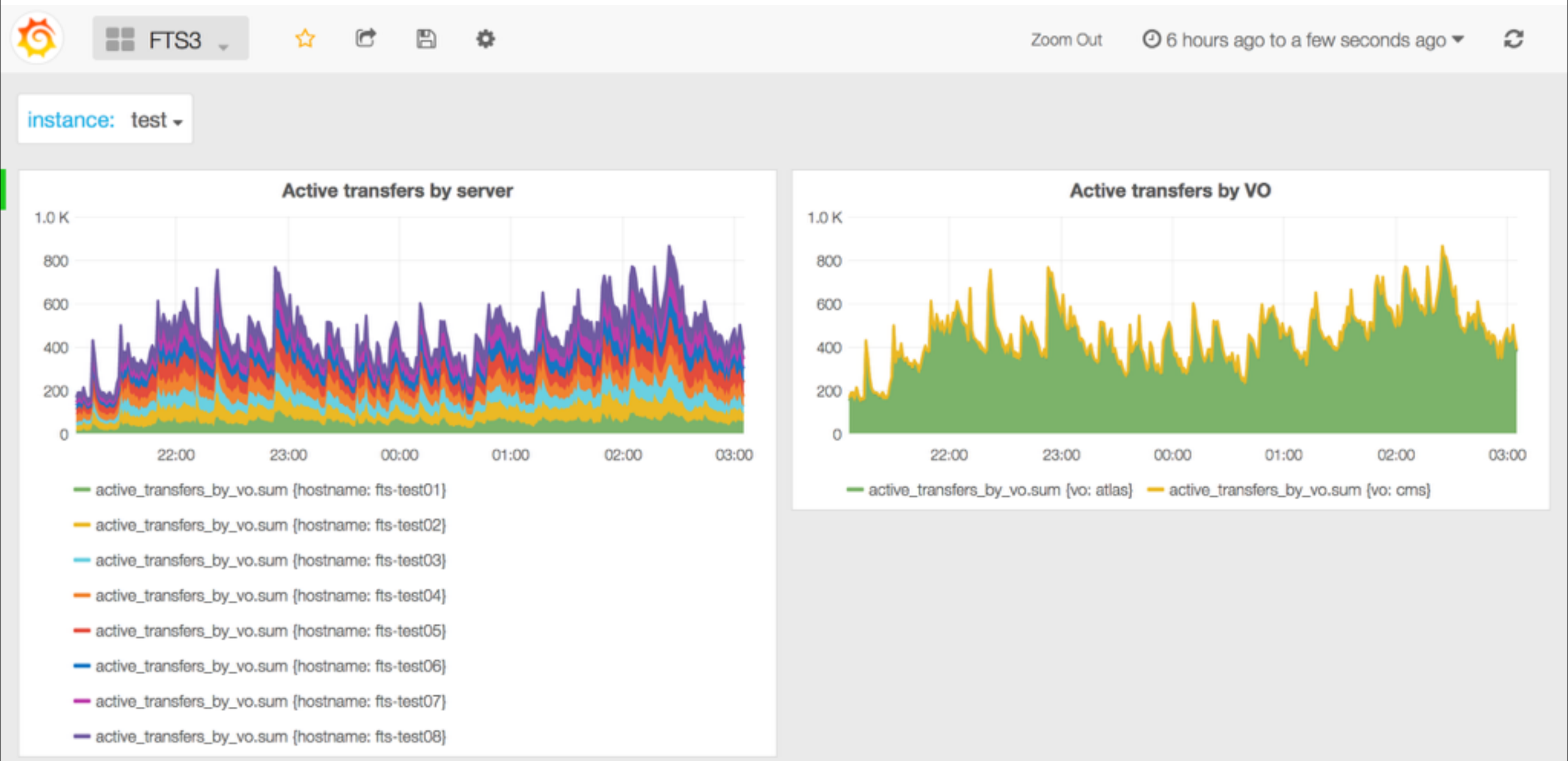
Grafana templating example - FTS3

- Can select between instances



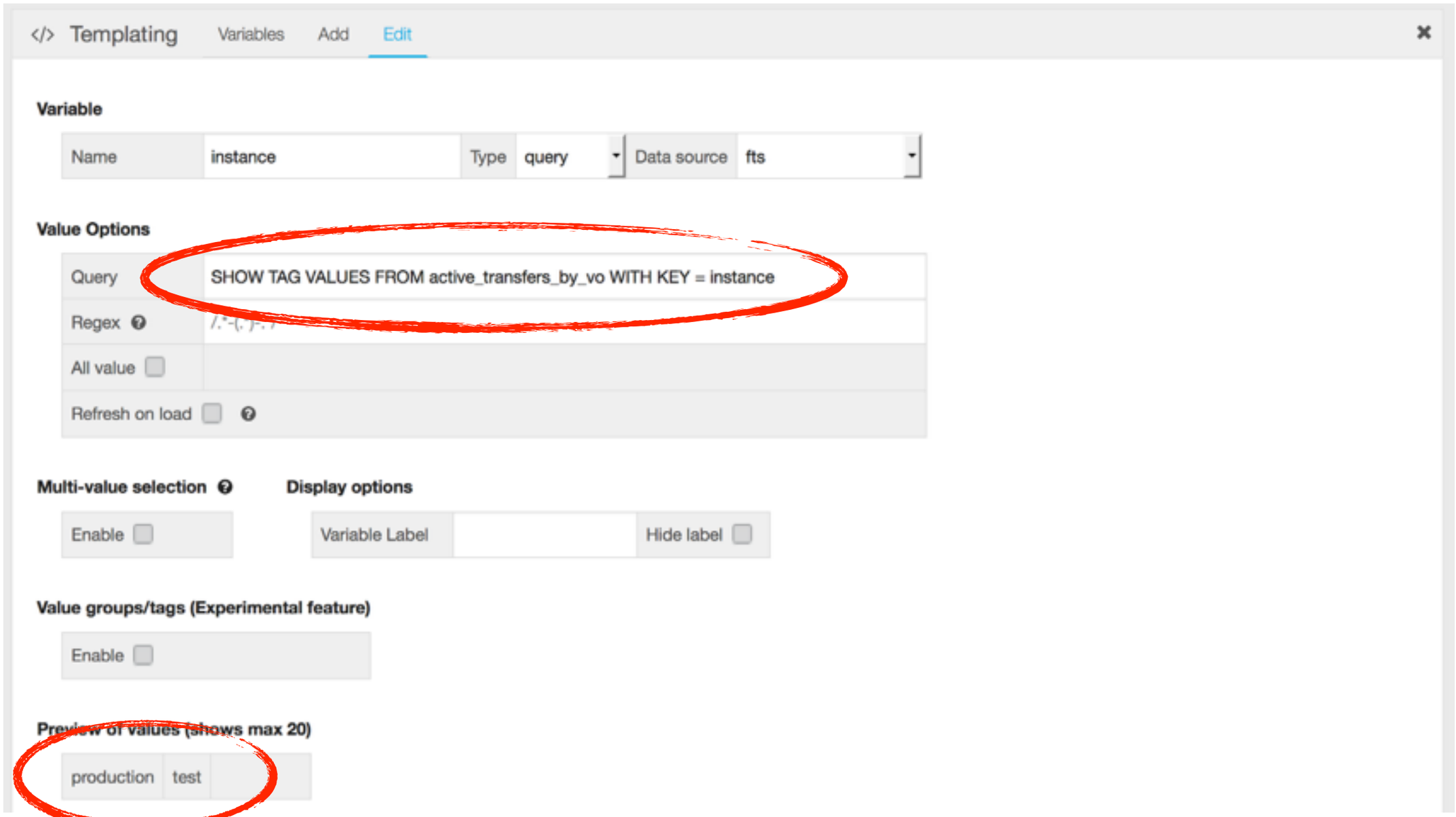
Grafana templating example - FTS3

- View of “test’ instance



Grafana templating example - FTS3

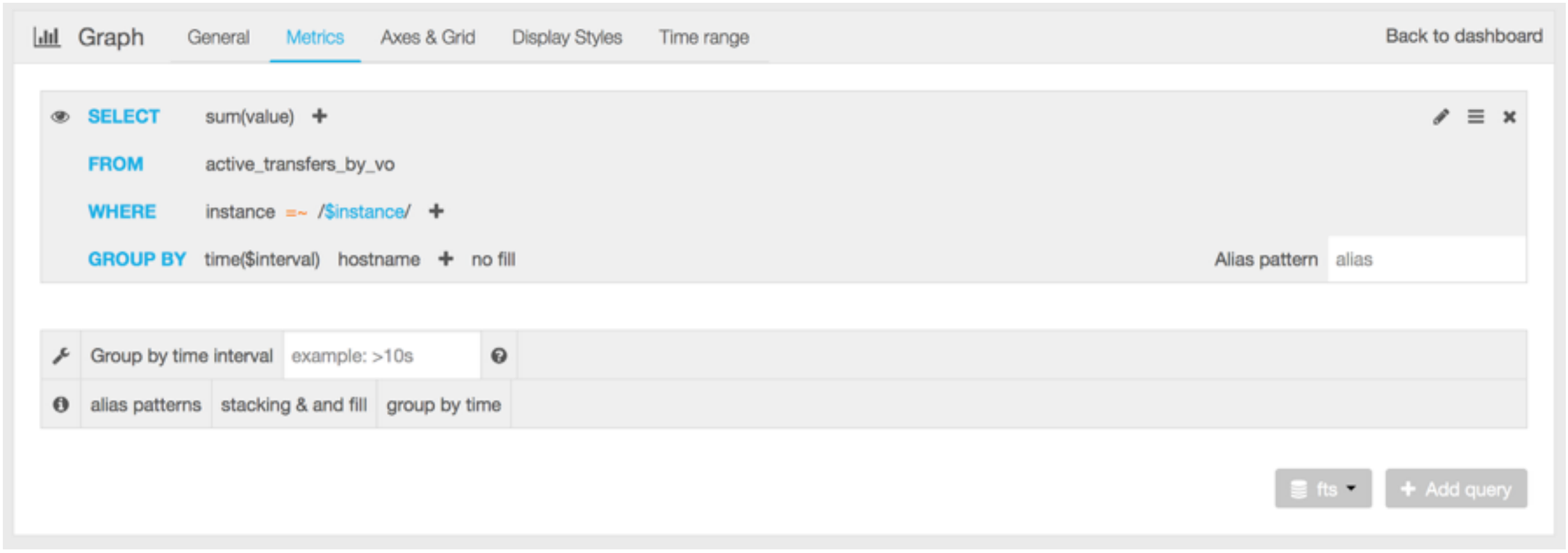
- Example - selecting different FTS3 instances



STFC

Grafana templating example - FTS3

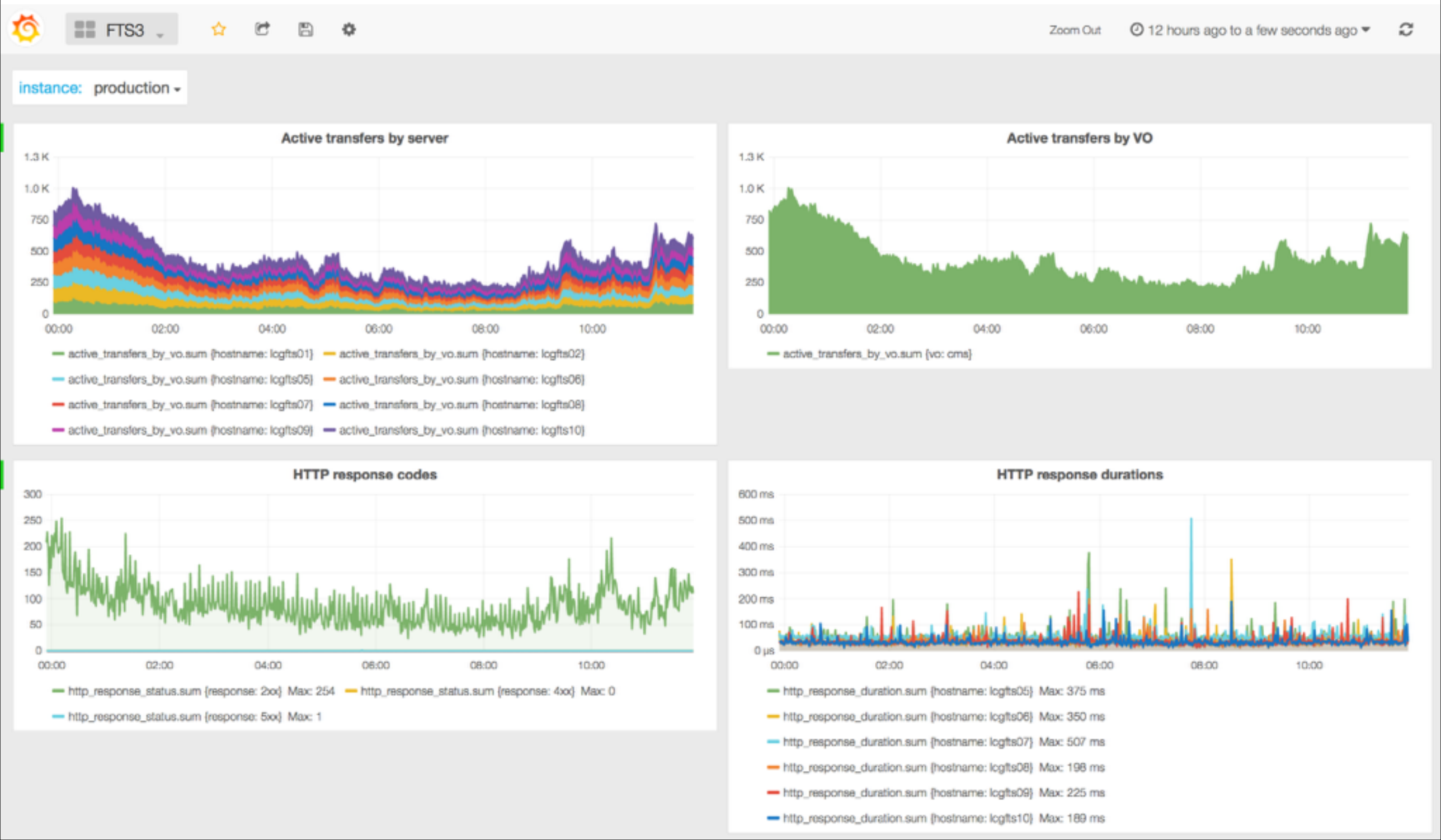
- Example - making an active transfers by hostname plot




```
active_transfers_by_vo,host=lcgfts01,instance=production,vo=atlas value=21
```

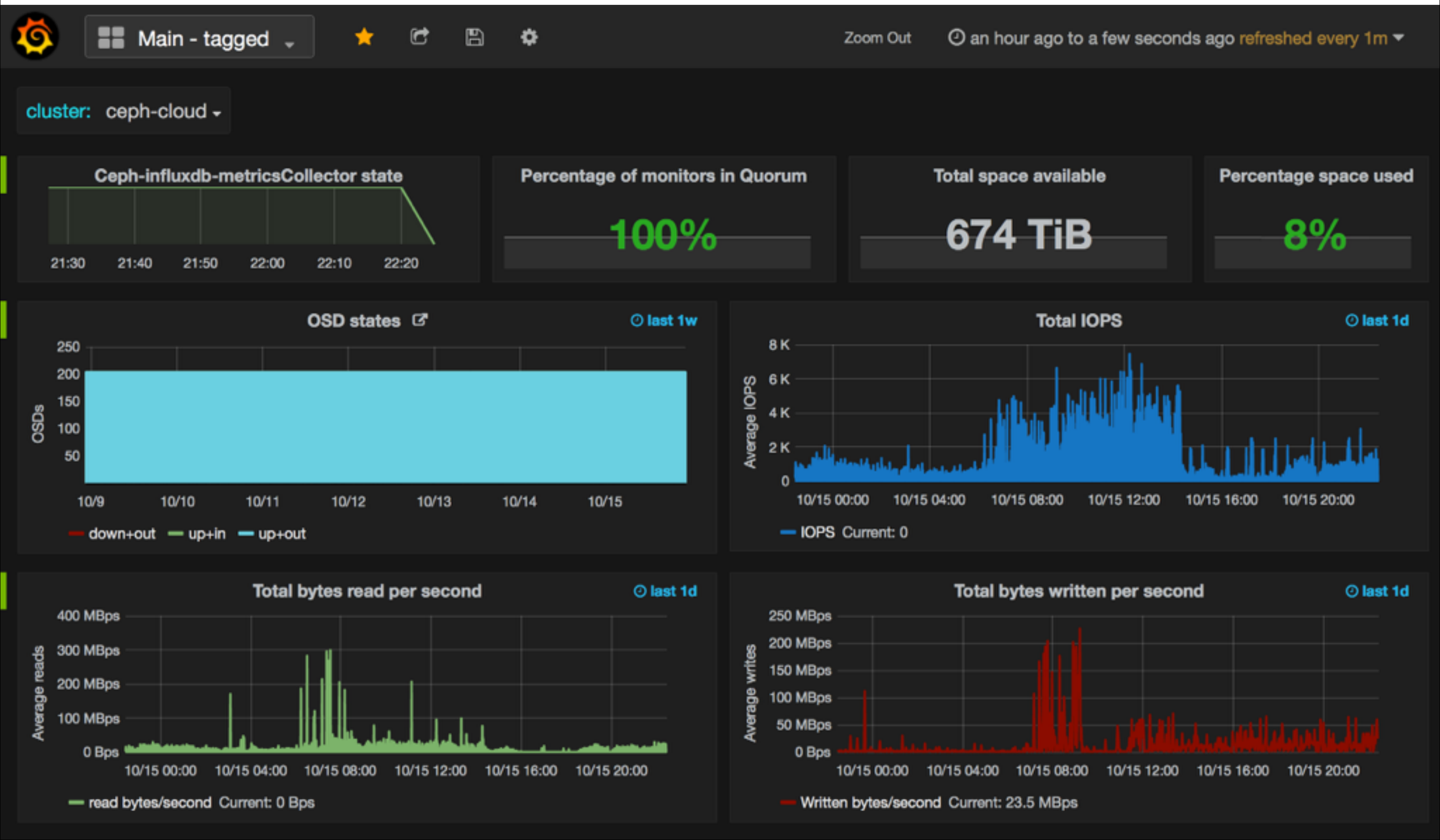
STFC

(Preliminary) FTS3 dashboard



Ceph dashboard

Ignacy Debicki
George Vasilakakos



InfluxDB data sources

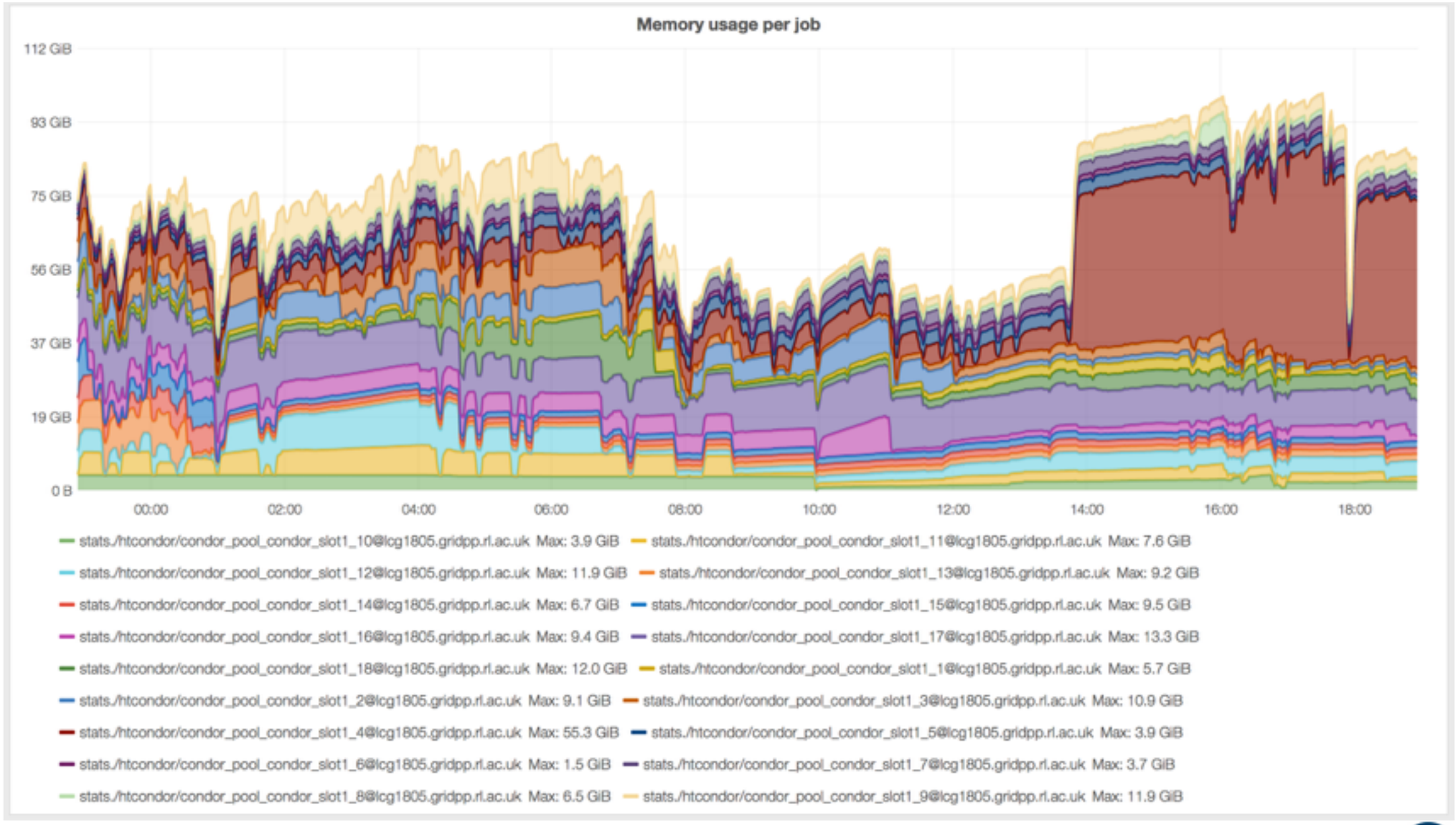
cAdvisor

- Container resource usage monitoring
  - Docker, cgroups (including HTCondor jobs)
  - metrics can be sent to InfluxDB (or Elasticsearch)
- Issues
  - only works with InfluxDB 0.8.x; waiting on https://github.com/google/cadvisor/pull/800
  - with default (dynamic) resolution, can be quite slow to make plots in Grafana

STFC

cAdvisor

- Example: memory usage per slot on a WN

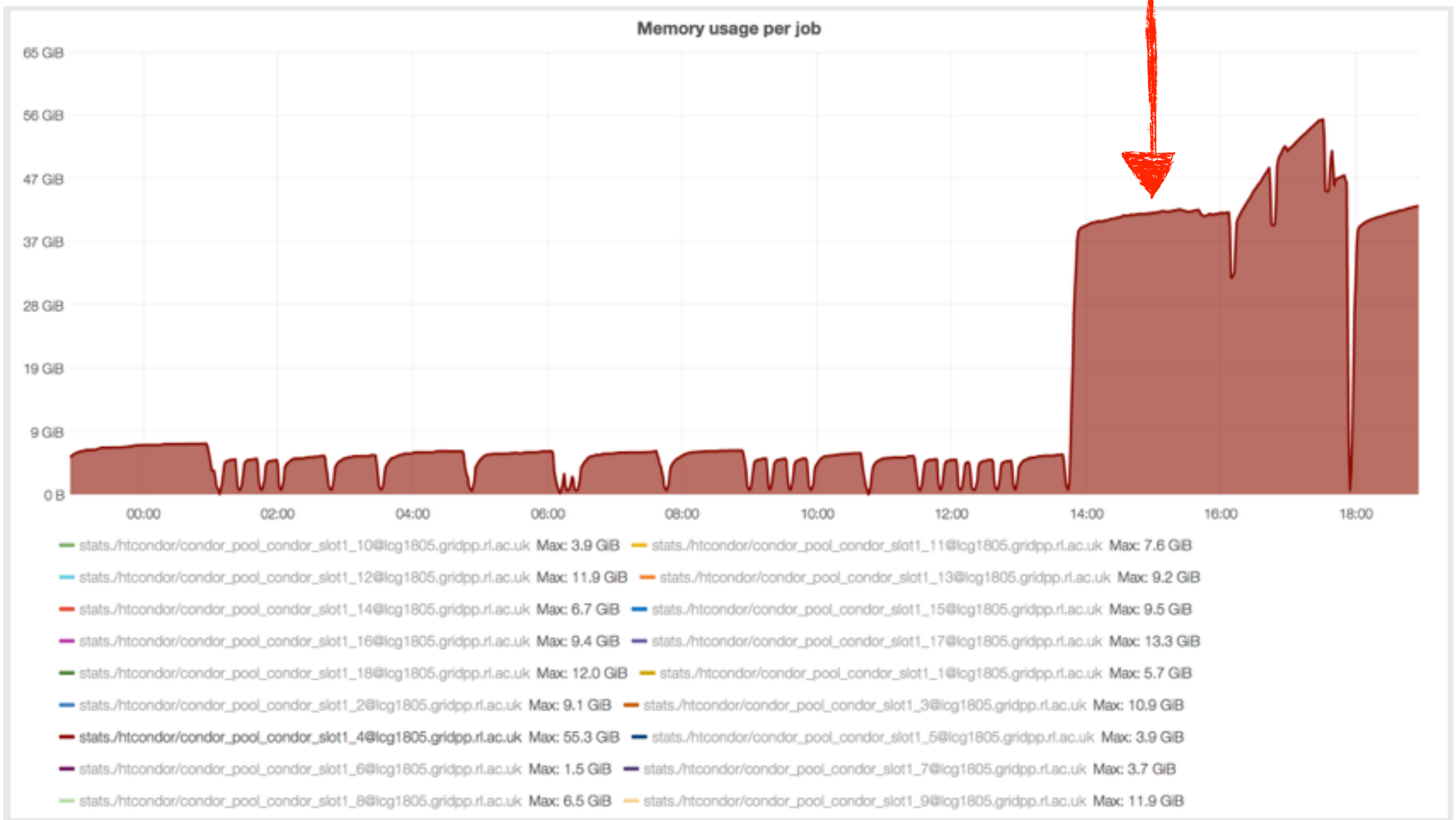


STFC

cAdvisor

- An interesting job…

ATLAS job (requested 16 GB memory)



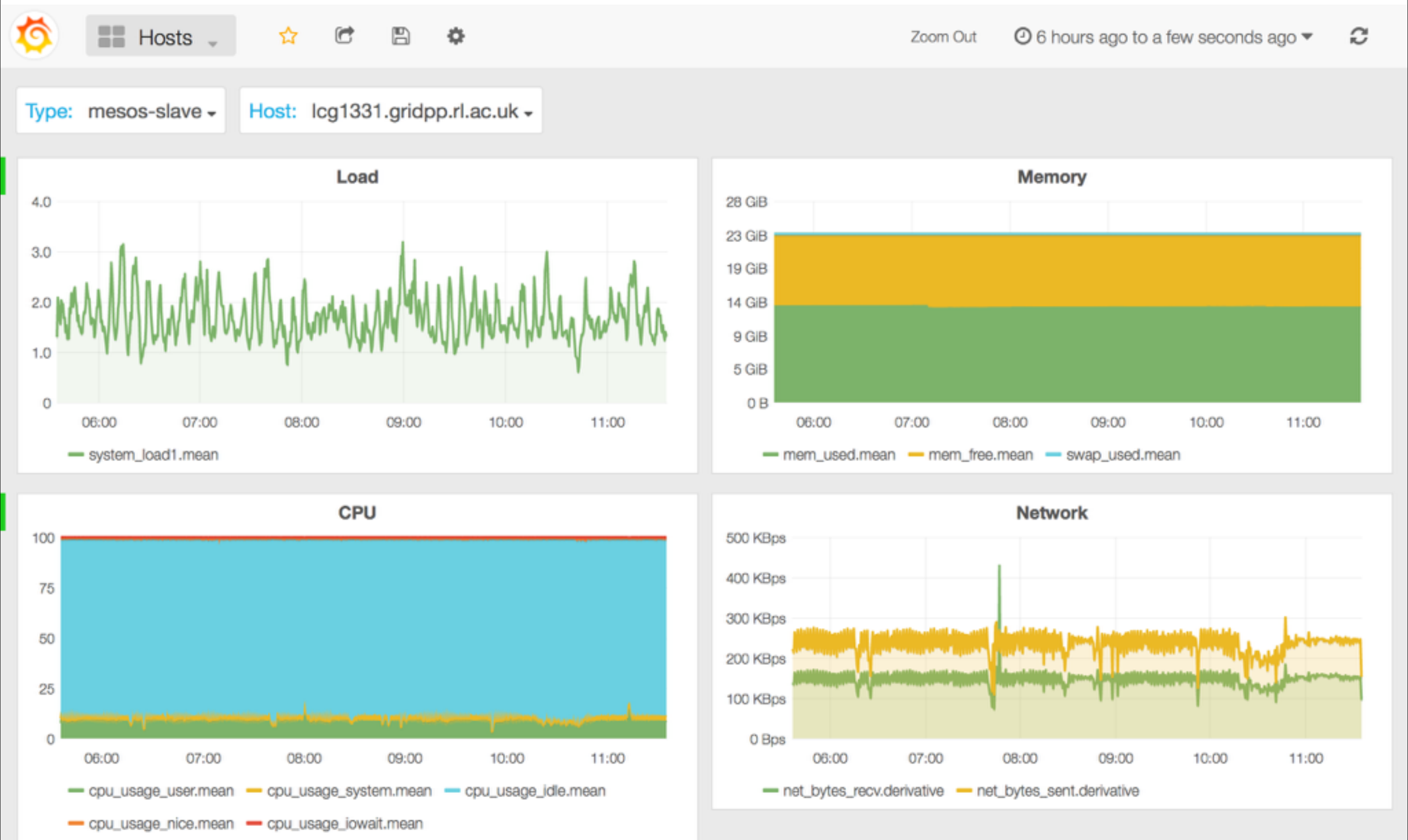
STFC

Telegraf

- Collects system and/or metrics from services, writes into InfluxDB
- System metrics
  - load, CPU, memory, network, disk IO, disk usage, swap, …
- Plugins for service specific metrics
  - Apache, MySQL, HAProxy, Elasticsearch, ZooKeeper, …
  - Can specify a script which produces metrics in json format
  - Write your own plugin… e.g. Ceph
- By default collects metrics every 10s, but this is configurable

STFC

Telegraf - basic host metrics



Telegraf running on many nodes

- What happens if Telegraf (or collectd, …) is running everywhere? Can a single InfluxDB node keep up?
- First test (last night)
  - InfluxDB 0.9.4 (running in a container on bare metal)
  - 189 Telegraf instances running (load, CPU, memory, network, disk metrics)
  - Telegraf sending metrics every 10s, with a 5s timeout configured
- Getting lots of errors like:

```
[write] 2015/10/16 13:10:26 write failed for shard 11 on node 1: engine: write points: write throughput too high. backoff and retry
```

- Also lots of HTTP 500 errors due to the 5s timeout
- **More investigation needed!**

STFC

Summary

- InfluxDB + Grafana make it easy to collect metrics & make nice useful dashboards
- Open questions
  - Best way to make publicly-accessible plots?
  - Can we replace Ganglia for system metrics on every machine?
    - stress-testing of InfluxDB needed, possibly a cluster is required
    - will present results at next HEPiX

STFC