Scalable Metrics Collection using Prometheus and Thanos

Jonathan Eichelberger and Shawn Hall
Lustre Webinar – Sept 9 2020

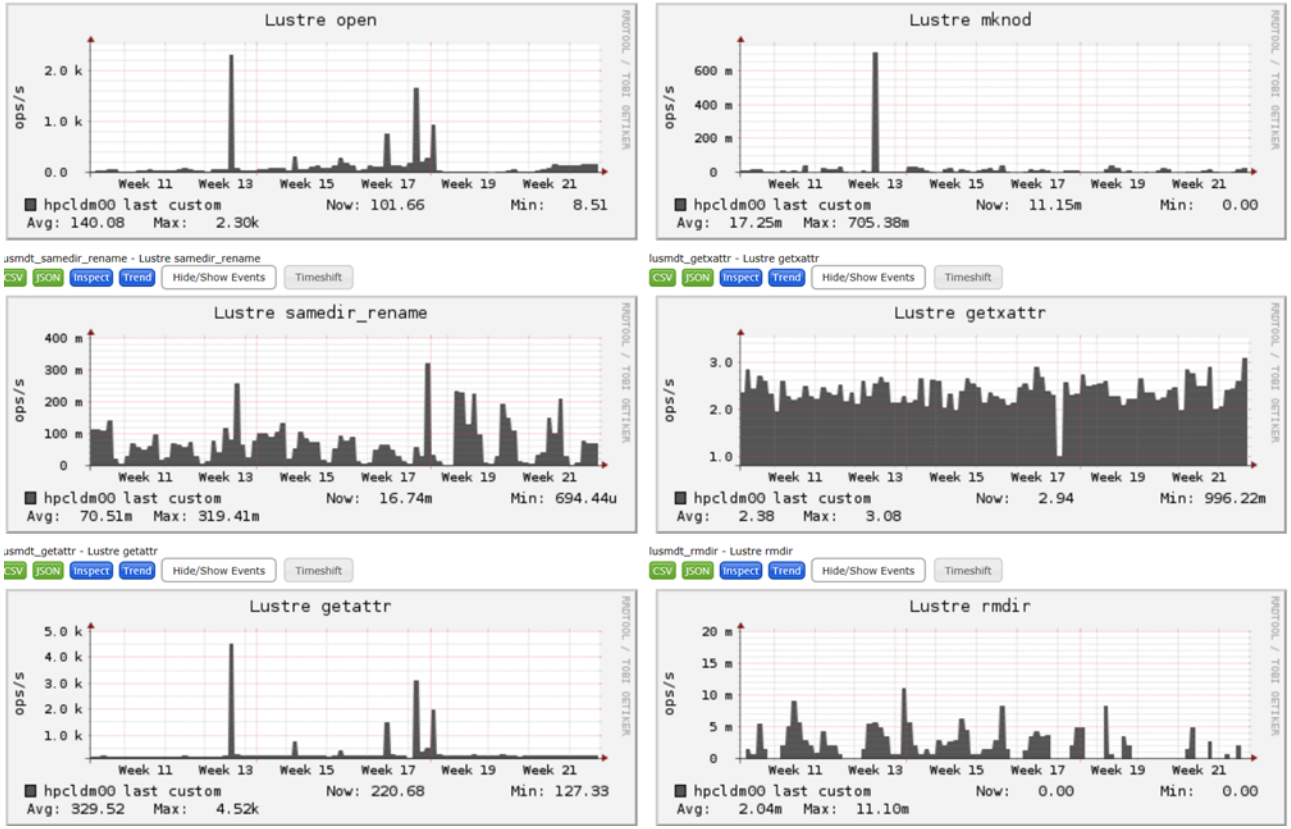
Metrics Collection Wishlist

- Scalable to meet our needs
- Easy to implement
- Easy to manage
- Reasonable storage requirements
- Can handle high cardinality
- Single pane of glass

History of Metrics Collection at BP

- Have used several toolkits
- Some were more successful than others
- Only the simplest or most useful have survived

Ganglia



```
Lustre open
■ hpcldm00 last custom          Now: 101.66       Min:    8.51
Avg: 140.08    Max:    2.30k

Lustre mknod
■ hpcldm00 last custom          Now:  11.15m      Min:    0.00
Avg:  17.25m   Max: 705.38m

Lustre samedir_rename
■ hpcldm00 last custom          Now:  16.74m      Min: 694.44u
Avg:  70.51m   Max: 319.41m

Lustre getxattr
■ hpcldm00 last custom          Now:   2.94       Min: 996.22m
Avg:   2.38    Max:    3.08

Lustre getattr
■ hpcldm00 last custom          Now: 220.68       Min: 127.33
Avg: 329.52    Max:    4.52k

Lustre rmdir
■ hpcldm00 last custom          Now:   0.00       Min:    0.00
Avg:   2.04m   Max:  11.10m
```

Ganglia

- Designed for clusters and grids
- Works well for aggregating cluster information into top level views
- RRD format works well for summaries, but inherently loses information
- Handles compute metrics, but required customization for Lustre monitoring
- Nothing was wrong, but our installation rotted away when the maintainer left our group

Splunk

- Awesome if you have a campus wide, unlimited license
- Not so awesome if you have to pay for a license yourself
- Shines at log collection and analysis, but also works well for metrics
- Never could show enough value to justify the cost

Telegraf + InfluxDB + Grafana

- Simple to install and configure
- Can parse Lustre jobstats
- Worked great initially, but…

Telegraf + InfluxDB + Grafana Problems

- No matter what time window you use, Grafana + InfluxDB should display an appropriate number of data points (1 hour window = 300 data points, 24 hour window = 300 data points)
  - But instead, amplitude was also scaled (1 hour window – 1 GB/s, 24 hour window - 24 GB/s)
  - Had to manually set resolution instead, meaning it was impossible to view data over large time windows

Telegraf + InfluxDB + Grafana Problems

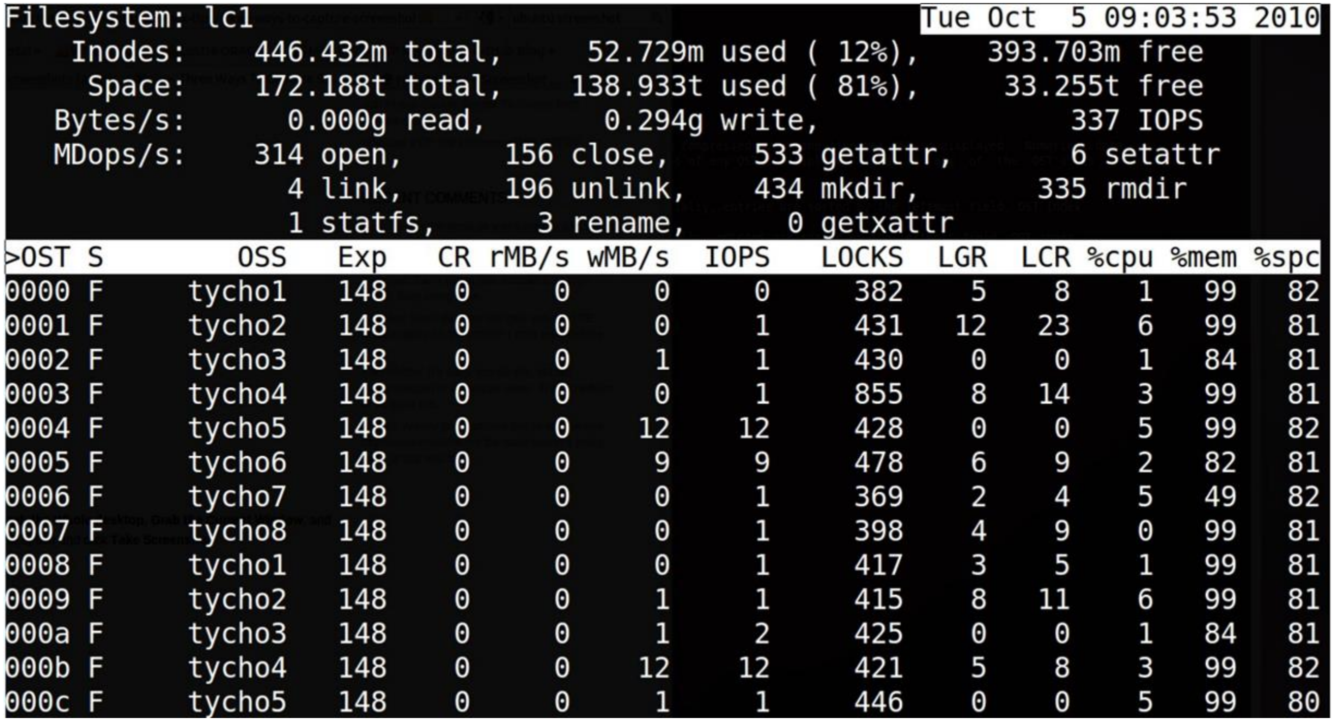
- If data resolution not fixed, huge spikes sometimes appear at beginning of graphs making them unreadable
  - Still no real fix – just workarounds - https://github.com/influxdata/influxdb/issues/6451
  - Issue is 4 years, 4 months, 18 days old today- but who's counting!

Telegraf + InfluxDB + Grafana Problems

- Jobstats cardinality kills InfluxDB
  - Function of a number of jobs, but we don’t have a ton of jobs
  - Horizontal scaling requires InfluxDB Enterprise
  - InfluxDB Enterprise requires money

Current Metrics Collection at BP

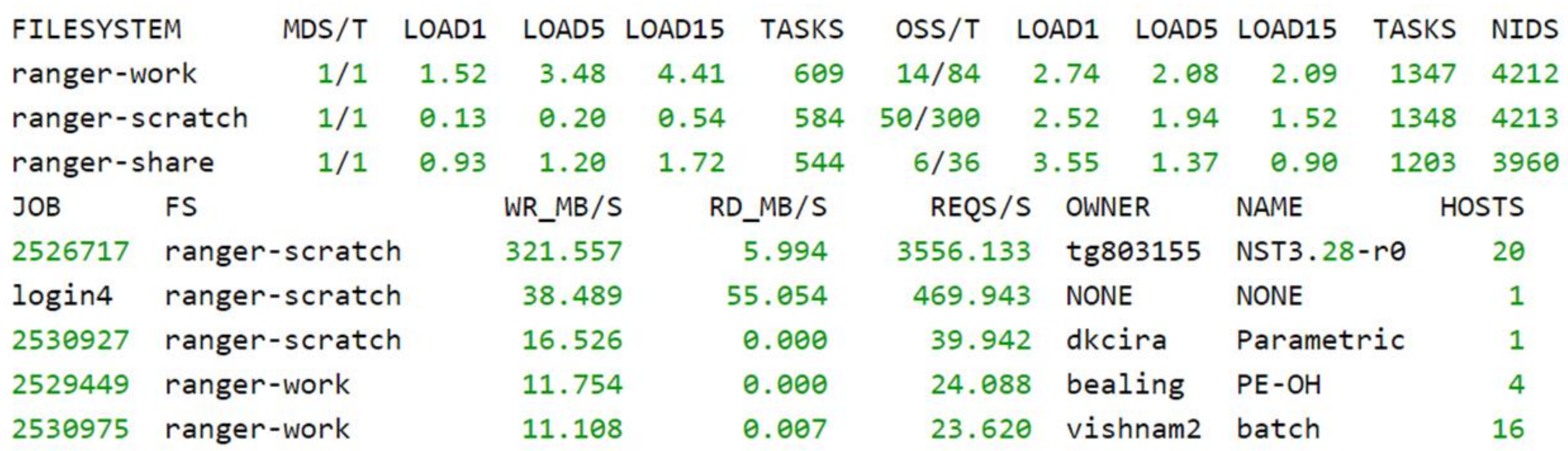
- Lustre Monitoring Tool
  - Condensed view of server-side Lustre activity
  - First thing we put on a new file system
  - No historical data



```
Filesystem: lc1                                        Tue Oct  5 09:03:53 2010
    Inodes:  446.432m total,     52.729m used ( 12%),    393.703m free
     Space:  172.188t total,    138.933t used ( 81%),     33.255t free
   Bytes/s:    0.000g read,       0.294g write,              337 IOPS
   MDops/s:    314 open,      156 close,     533 getattr,      6 setattr
                 4 link,      196 unlink,    434 mkdir,      335 rmdir
                 1 statfs,      3 rename,      0 getxattr
>OST S         OSS    Exp    CR rMB/s wMB/s  IOPS   LOCKS  LGR  LCR %cpu %mem %spc
0000 F      tycho1    148     0     0     0     0     382    5    8    1   99   82
0001 F      tycho2    148     0     0     0     1     431   12   23    6   99   81
0002 F      tycho3    148     0     0     1     1     430    0    0    1   84   81
0003 F      tycho4    148     0     0     0     1     855    8   14    3   99   81
0004 F      tycho5    148     0     0    12    12     428    0    0    5   99   82
0005 F      tycho6    148     0     0     9     9     478    6    9    2   82   81
0006 F      tycho7    148     0     0     0     1     369    2    4    5   49   82
0007 F      tycho8    148     0     0     0     1     398    4    9    0   99   81
0008 F      tycho1    148     0     0     0     1     417    3    5    1   99   81
0009 F      tycho2    148     0     0     1     1     415    8   11    6   99   81
000a F      tycho3    148     0     0     1     2     425    0    0    1   84   81
000b F      tycho4    148     0     0    12    12     421    5    8    3   99   82
000c F      tycho5    148     0     0     1     1     446    0    0    5   99   80
```

Current Metrics Collection at BP

- xltop
  - Gives critical relationship between jobs and file system performance
  - No historical data
  - TACC’s updates aren’t publicly available - we’re using 2012 code ☹



```
FILESYSTEM        MDS/T  LOAD1  LOAD5 LOAD15  TASKS   OSS/T  LOAD1  LOAD5 LOAD15  TASKS  NIDS
ranger-work         1/1   1.52   3.48   4.41    609   14/84   2.74   2.08   2.09   1347  4212
ranger-scratch      1/1   0.13   0.20   0.54    584  50/300   2.52   1.94   1.52   1348  4213
ranger-share        1/1   0.93   1.20   1.72    544    6/36   3.55   1.37   0.90   1203  3960
JOB      FS                WR_MB/S      RD_MB/S       REQS/S  OWNER     NAME         HOSTS
2526717  ranger-scratch    321.557        5.994     3556.133  tg803155  NST3.28-r0      20
login4   ranger-scratch     38.489       55.054      469.943  NONE      NONE             1
2530927  ranger-scratch     16.526        0.000       39.942  dkcira    Parametric       1
2529449  ranger-work        11.754        0.000       24.088  bealing   PE-OH            4
2530975  ranger-work        11.108        0.007       23.620  vishnam2  batch           16
```

Prometheus

- Prometheus is a pull-based metric collecting / monitoring framework.
  - a multi-dimensional data model (timeseries defined by metric name and set of key/value dimensions)
  - a flexible query language to leverage this dimensionality
  - no dependency on distributed storage; single server nodes are autonomous
  - timeseries collection happens via a pull model over HTTP
  - pushing timeseries is supported via an intermediary gateway
  - targets are discovered via service discovery or static configuration
  - multiple modes of graphing and dashboarding support

https://github.com/prometheus/prometheus

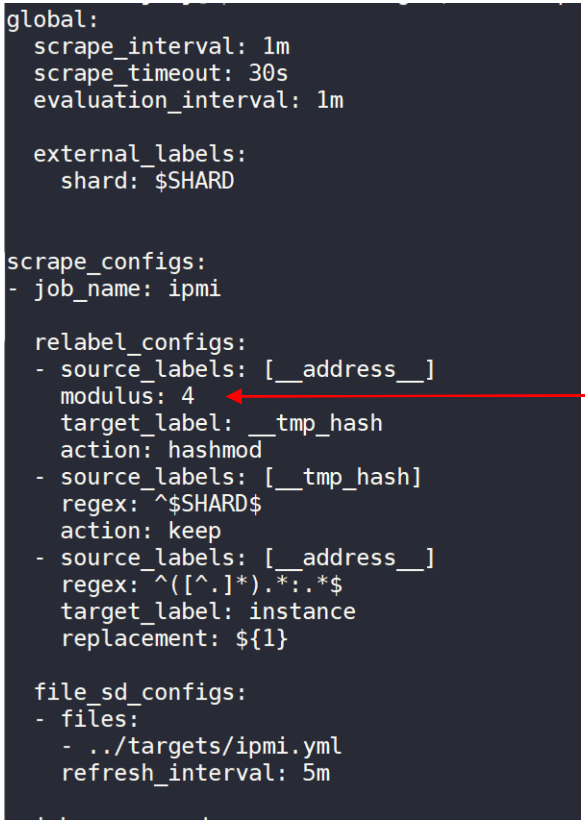
Thanos

- Thanos is a helper framework that allows Prometheus to be a highly available and scalable solution for monitoring large datacenters.
  - Global querying view across all connected Prometheus servers
  - Deduplication and merging of metrics collected from Prometheus HA pairs
  - Seamless integration with existing Prometheus setups
  - Downsampling historical data for massive query speedup
  - Simple gRPC "Store API" for unified data access across all metric data

https://github.com/thanos-io/thanos

Easily Add More Prometheus Servers

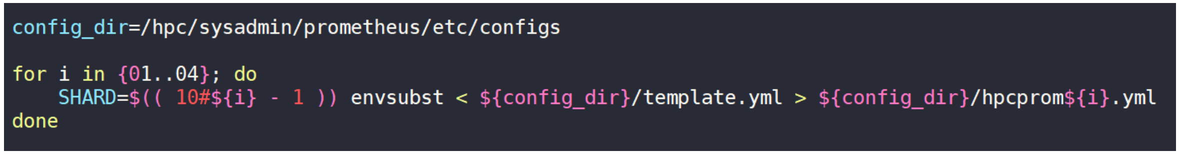
template.yml



```
global:
  scrape_interval: 1m
  scrape_timeout: 30s
  evaluation_interval: 1m

  external_labels:
    shard: $SHARD


scrape_configs:
- job_name: ipmi

  relabel_configs:
  - source_labels: [__address__]
    modulus: 4
    target_label: __tmp_hash
    action: hashmod
  - source_labels: [__tmp_hash]
    regex: ^$SHARD$
    action: keep
  - source_labels: [__address__]
    regex: ^([^.]*).*:.*$
    target_label: instance
    replacement: ${1}

  file_sd_configs:
  - files:
    - ../targets/ipmi.yml
    refresh_interval: 5m
```

Number of Prometheus servers

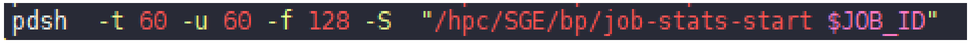
generate_configs.sh



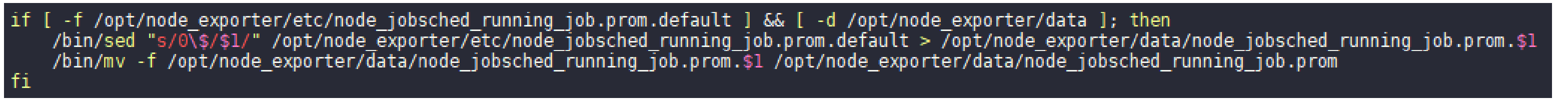
```
config_dir=/hpc/sysadmin/prometheus/etc/configs

for i in {01..04}; do
    SHARD=$(( 10#${i} - 1 )) envsubst < ${config_dir}/template.yml > ${config_dir}/hpcprom${i}.yml
done
```

Job Scheduler Integration

- In order to associate jobs with host metrics, a "flag" needs to be set on all compute nodes for the associated job.
  - Prolog

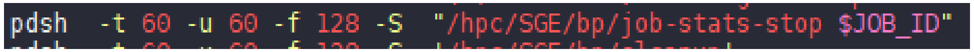


```
pdsh  -t 60 -u 60 -f 128 -S  "/hpc/SGE/bp/job-stats-start $JOB_ID"
```

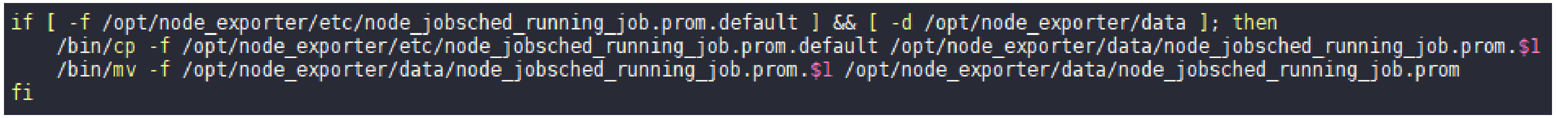


```
if [ -f /opt/node_exporter/etc/node_jobsched_running_job.prom.default ] && [ -d /opt/node_exporter/data ]; then
    /bin/sed "s/0\$/$1/" /opt/node_exporter/etc/node_jobsched_running_job.prom.default > /opt/node_exporter/data/node_jobsched_running_job.prom.$1
    /bin/mv -f /opt/node_exporter/data/node_jobsched_running_job.prom.$1 /opt/node_exporter/data/node_jobsched_running_job.prom
fi
```

  - Epilog

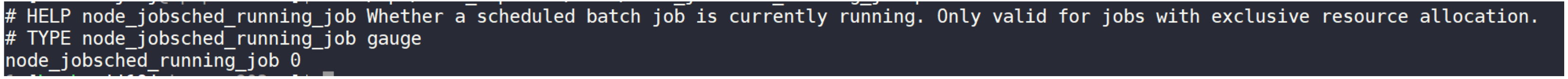


```
pdsh  -t 60 -u 60 -f 128 -S  "/hpc/SGE/bp/job-stats-stop $JOB_ID"
```



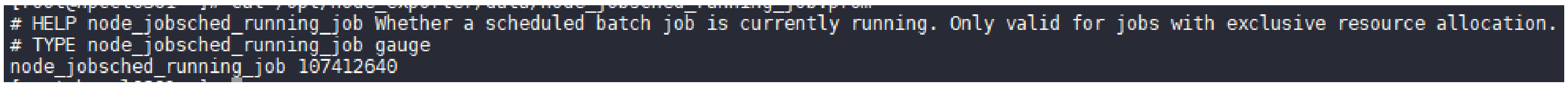
```
if [ -f /opt/node_exporter/etc/node_jobsched_running_job.prom.default ] && [ -d /opt/node_exporter/data ]; then
    /bin/cp -f /opt/node_exporter/etc/node_jobsched_running_job.prom.default /opt/node_exporter/data/node_jobsched_running_job.prom.$1
    /bin/mv -f /opt/node_exporter/data/node_jobsched_running_job.prom.$1 /opt/node_exporter/data/node_jobsched_running_job.prom
fi
```

Without job running



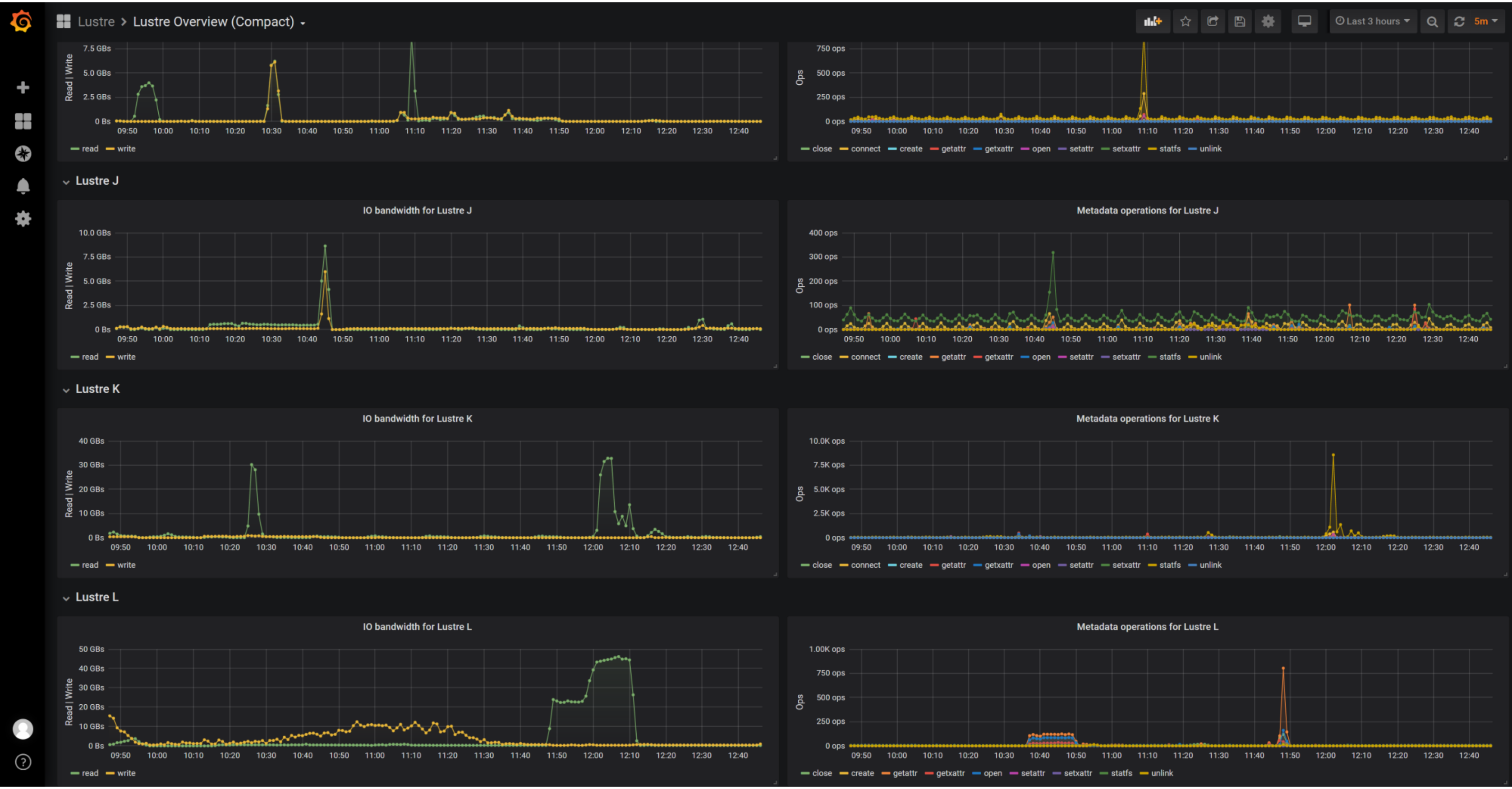
```
# HELP node_jobsched_running_job Whether a scheduled batch job is currently running. Only valid for jobs with exclusive resource allocation.
# TYPE node_jobsched_running_job gauge
node_jobsched_running_job 0
```

With job running

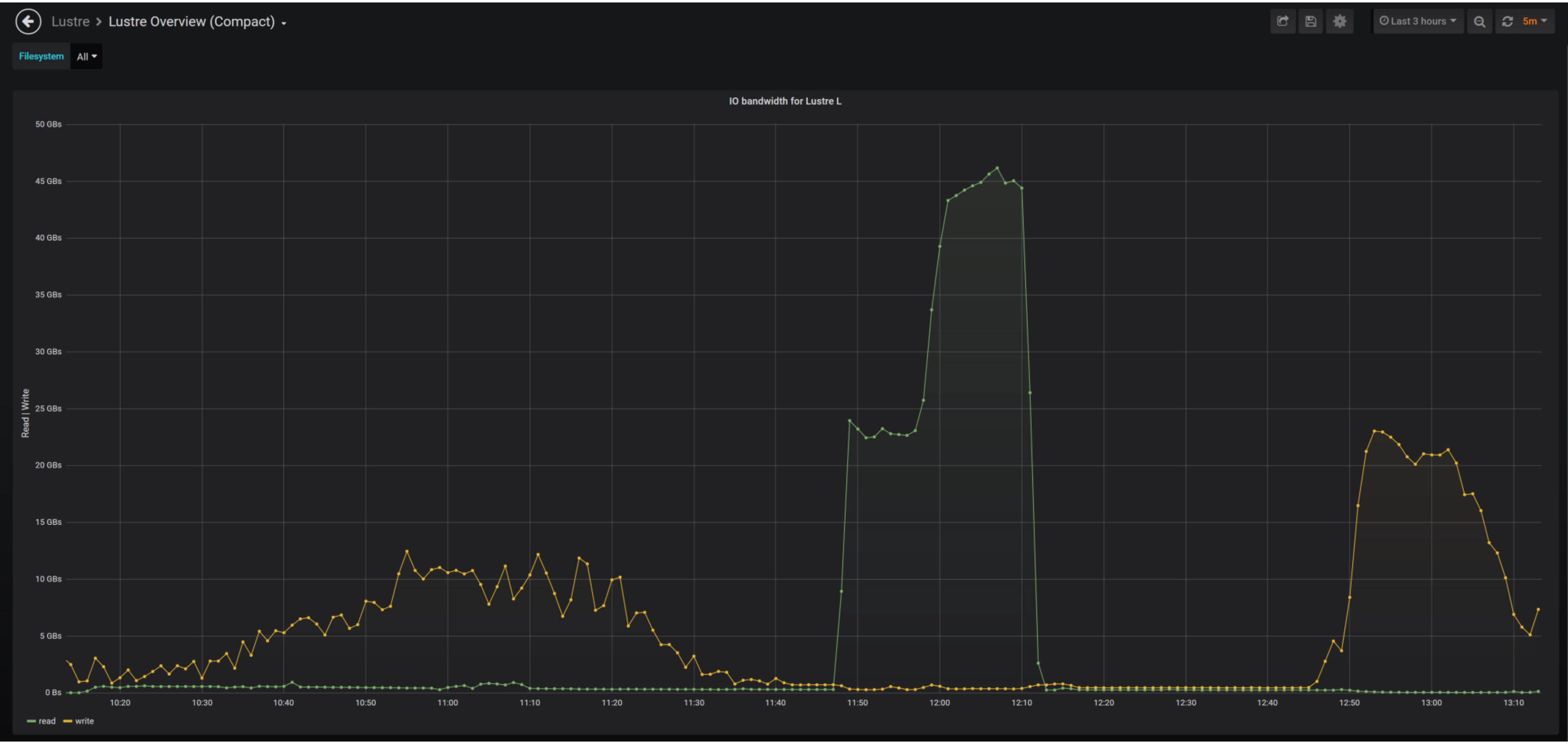


```
# HELP node_jobsched_running_job Whether a scheduled batch job is currently running. Only valid for jobs with exclusive resource allocation.
# TYPE node_jobsched_running_job gauge
node_jobsched_running_job 107412640
```

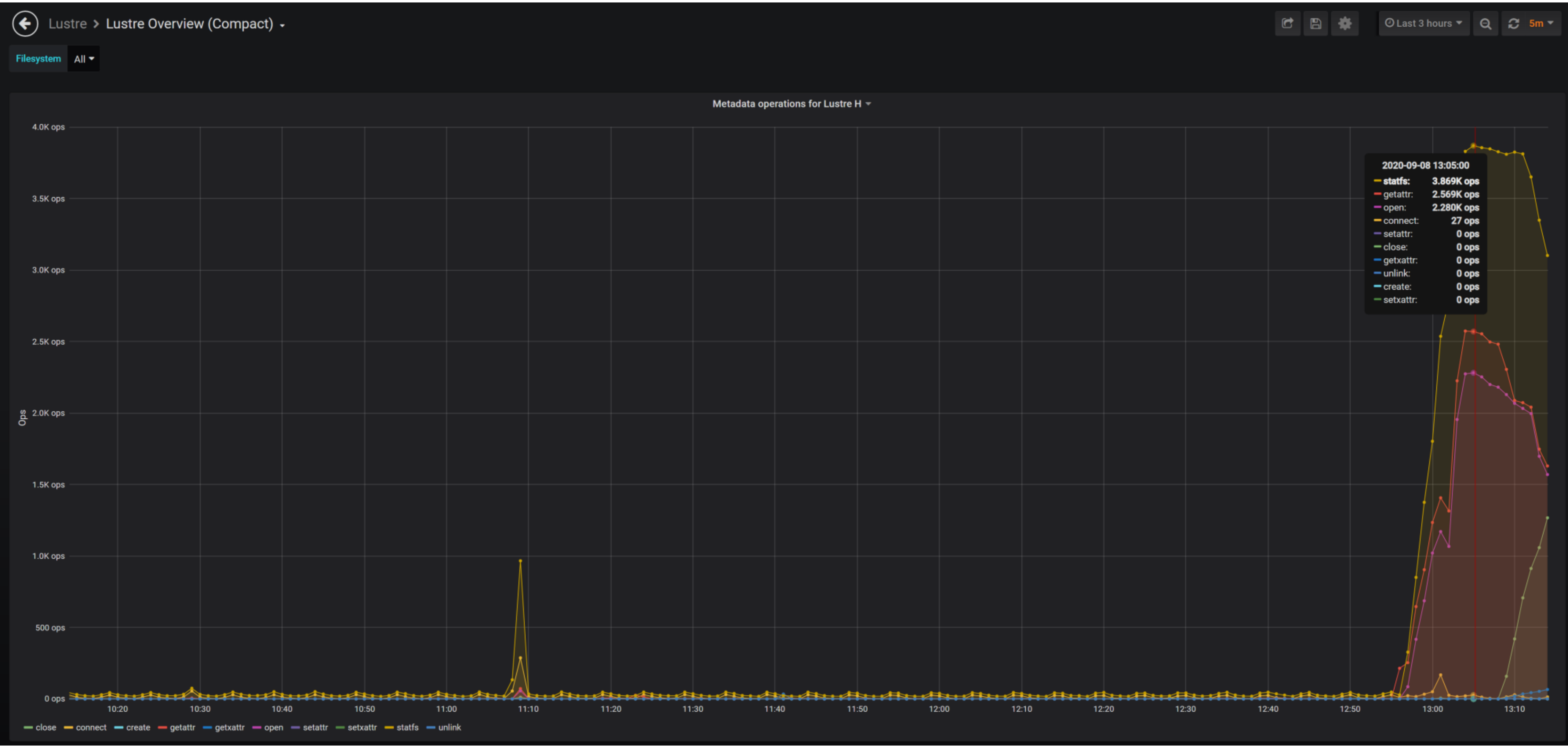
Lustre Overview Dashboard



Lustre Overview Dashboard



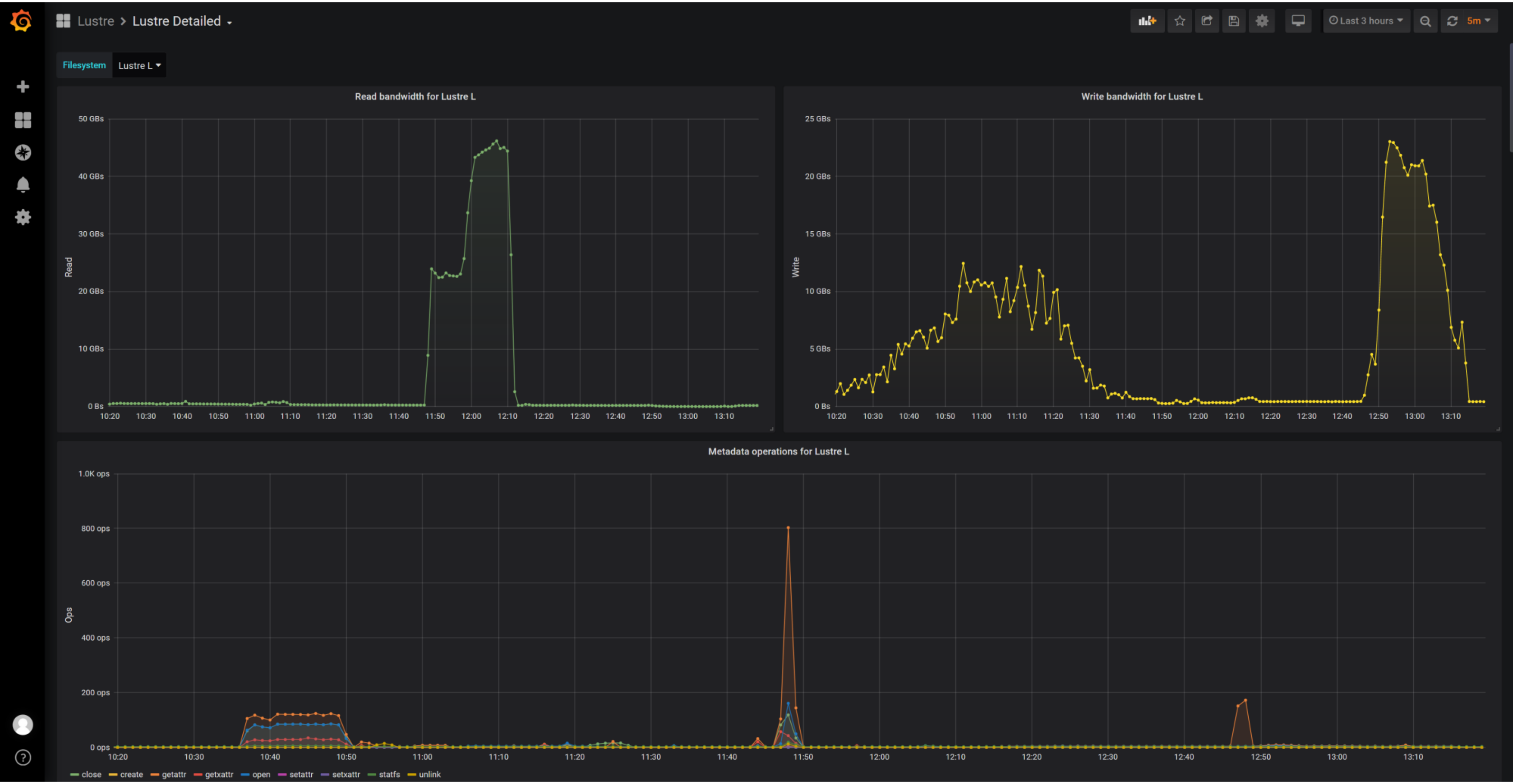
Lustre Overview Dashboard



2020-09-08 13:05:00

```
statfs:     3.869K ops
getattr:    2.569K ops
open:       2.280K ops
connect:        27 ops
setattr:         0 ops
close:           0 ops
getxattr:        0 ops
unlink:          0 ops
create:          0 ops
setxattr:        0 ops
```

Lustre Detail Dashboard



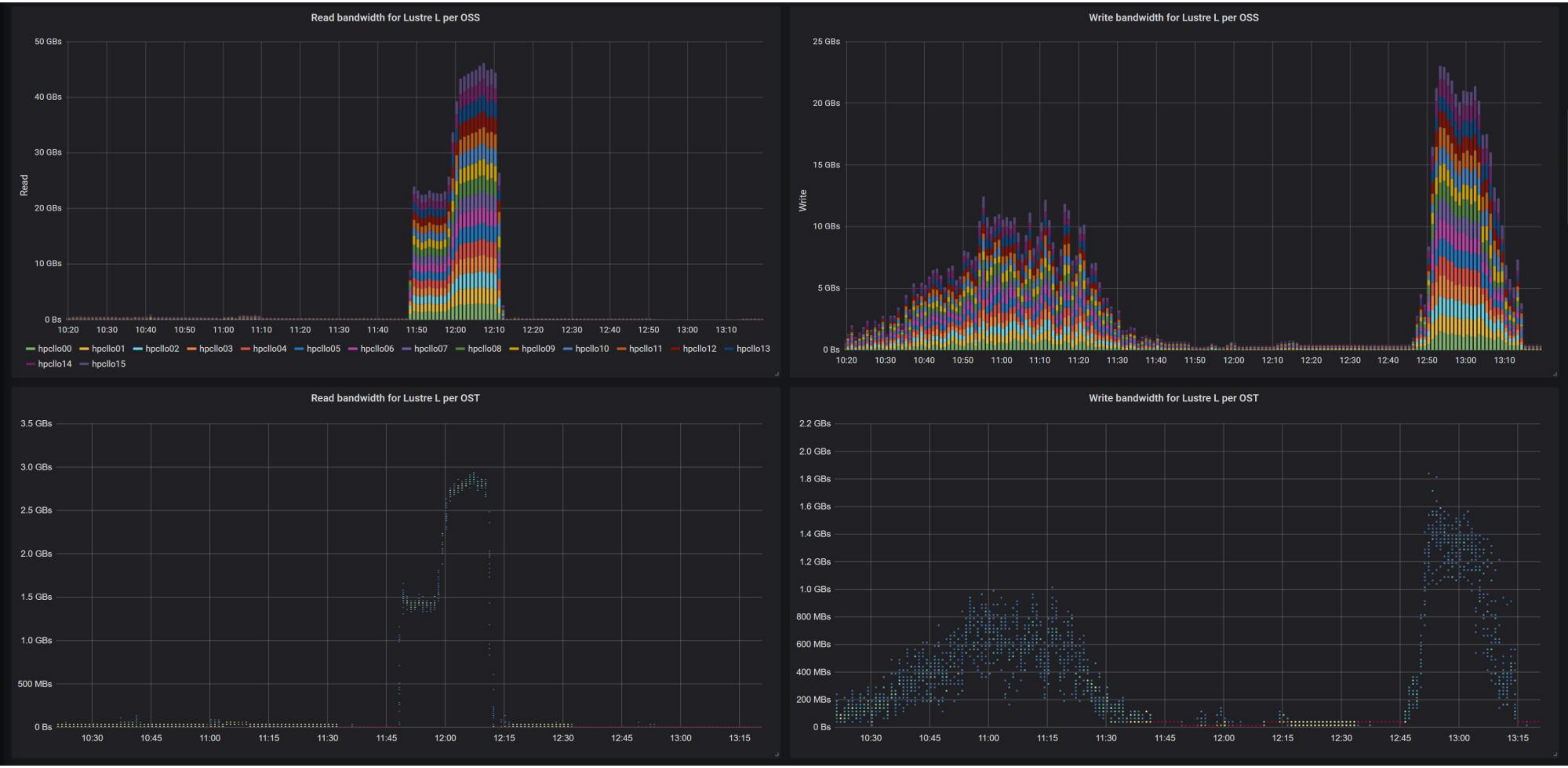
Lustre > Lustre Detailed

Filesystem Lustre L

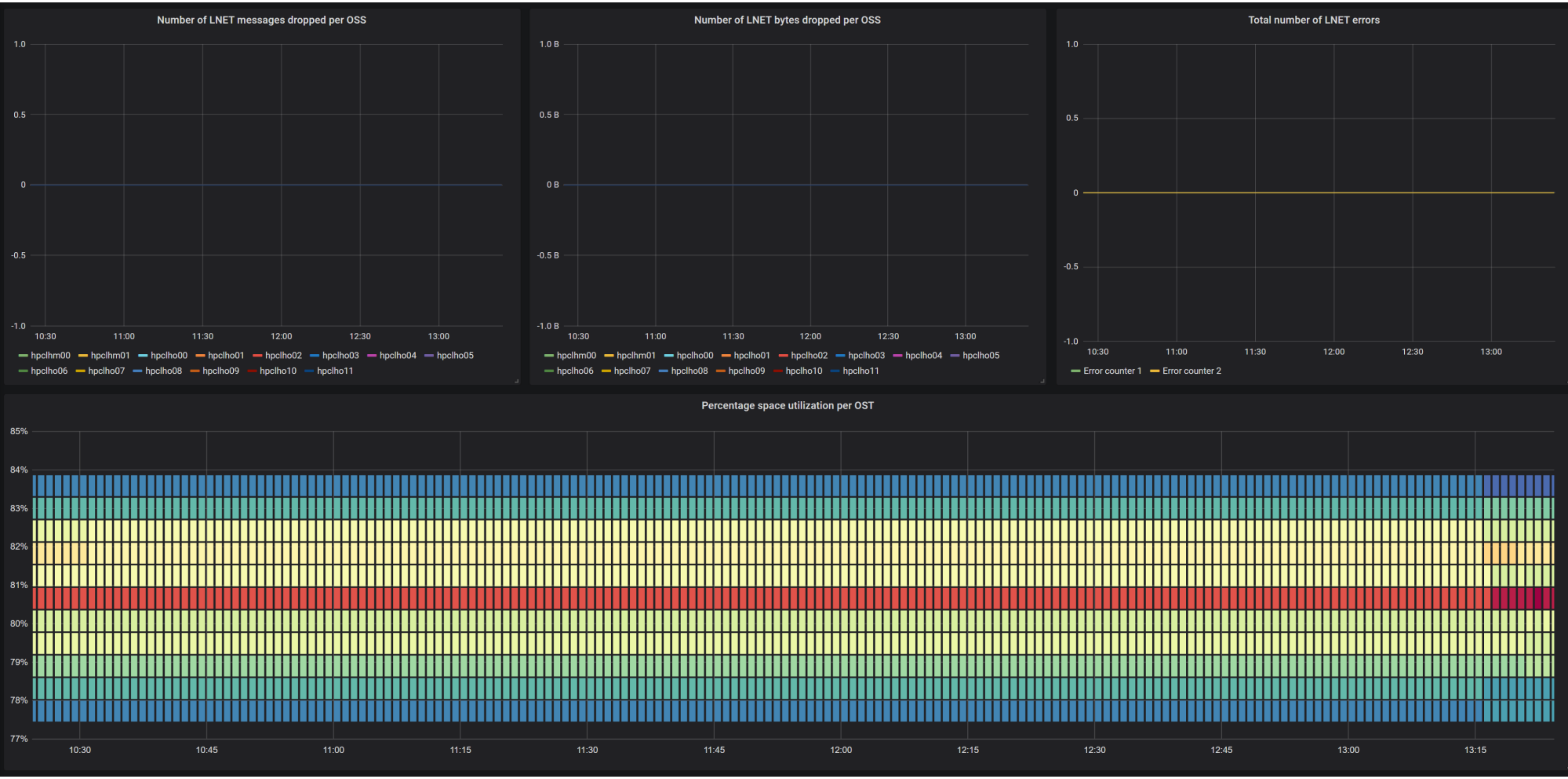
Last 3 hours

5m

Lustre Detail Dashboard



Lustre Detail Dashboard



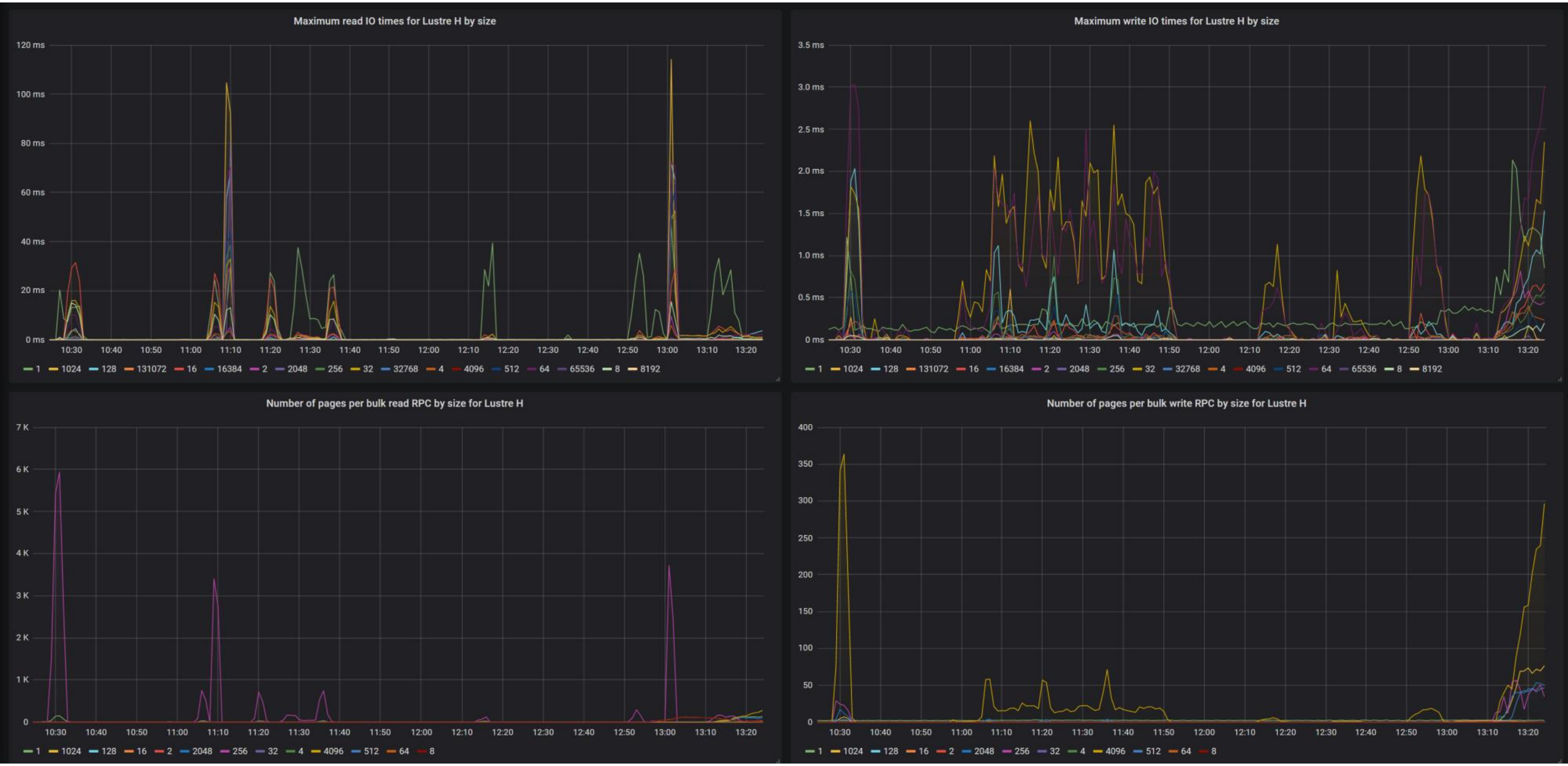
Number of LNET messages dropped per OSS

Number of LNET bytes dropped per OSS

Total number of LNET errors

Percentage space utilization per OST

Lustre Detail Dashboard



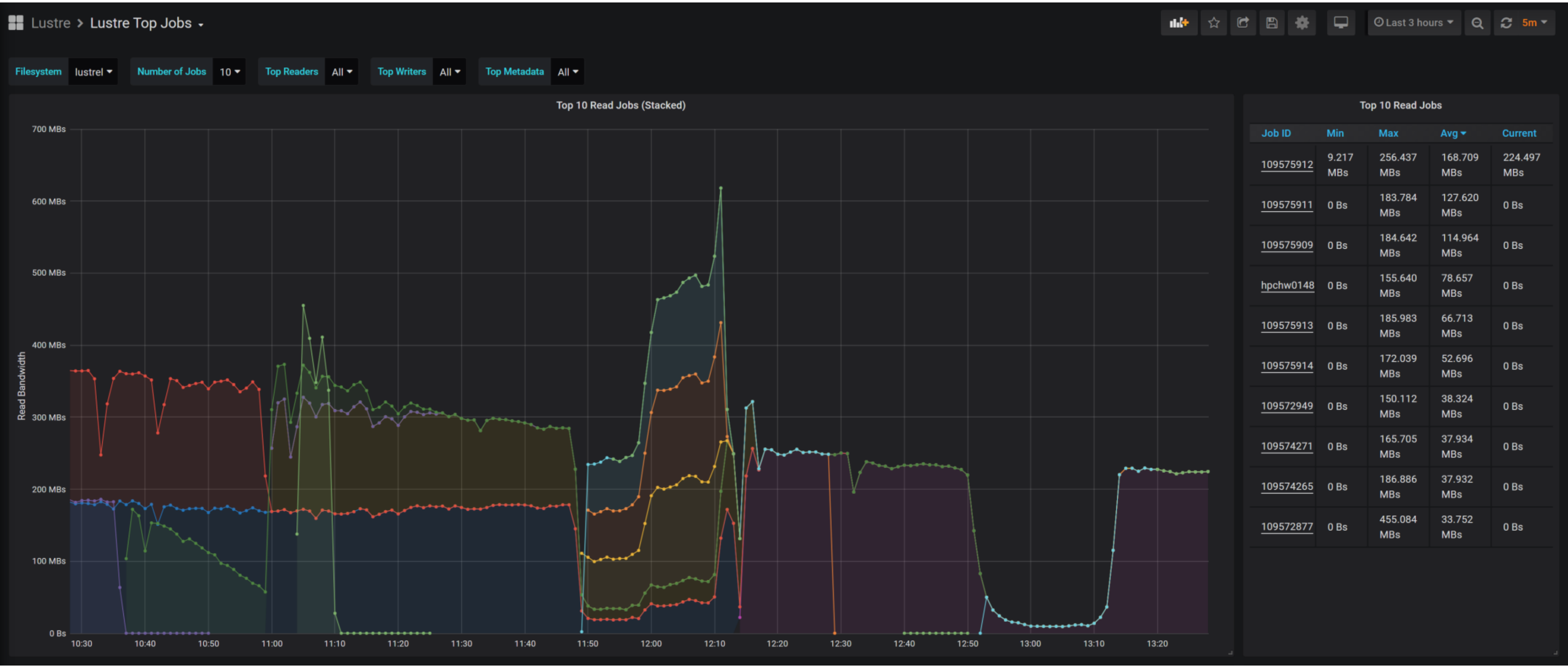
Maximum read IO times for Lustre H by size

Maximum write IO times for Lustre H by size

Number of pages per bulk read RPC by size for Lustre H

Number of pages per bulk write RPC by size for Lustre H

Lustre Top Jobs Dashboard



Lustre > Lustre Top Jobs

Filesystem: lustrel | Number of Jobs: 10 | Top Readers: All | Top Writers: All | Top Metadata: All

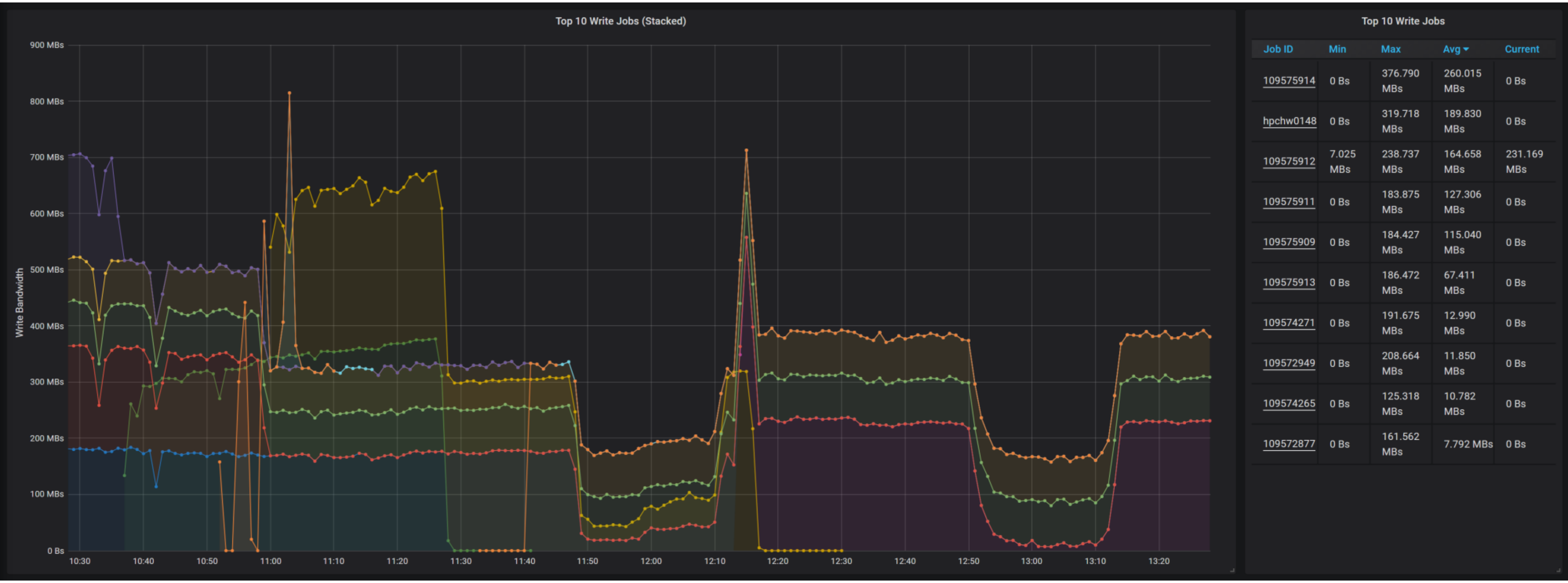
Last 3 hours | 5m

Top 10 Read Jobs (Stacked)

Top 10 Read Jobs

Job ID	Min	Max	Avg	Current
109575912	9.217 MBs	256.437 MBs	168.709 MBs	224.497 MBs
109575911	0 Bs	183.784 MBs	127.620 MBs	0 Bs
109575909	0 Bs	184.642 MBs	114.964 MBs	0 Bs
hpchw0148	0 Bs	155.640 MBs	78.657 MBs	0 Bs
109575913	0 Bs	185.983 MBs	66.713 MBs	0 Bs
109575914	0 Bs	172.039 MBs	52.696 MBs	0 Bs
109572949	0 Bs	150.112 MBs	38.324 MBs	0 Bs
109574271	0 Bs	165.705 MBs	37.934 MBs	0 Bs
109574265	0 Bs	186.886 MBs	37.932 MBs	0 Bs
109572877	0 Bs	455.084 MBs	33.752 MBs	0 Bs

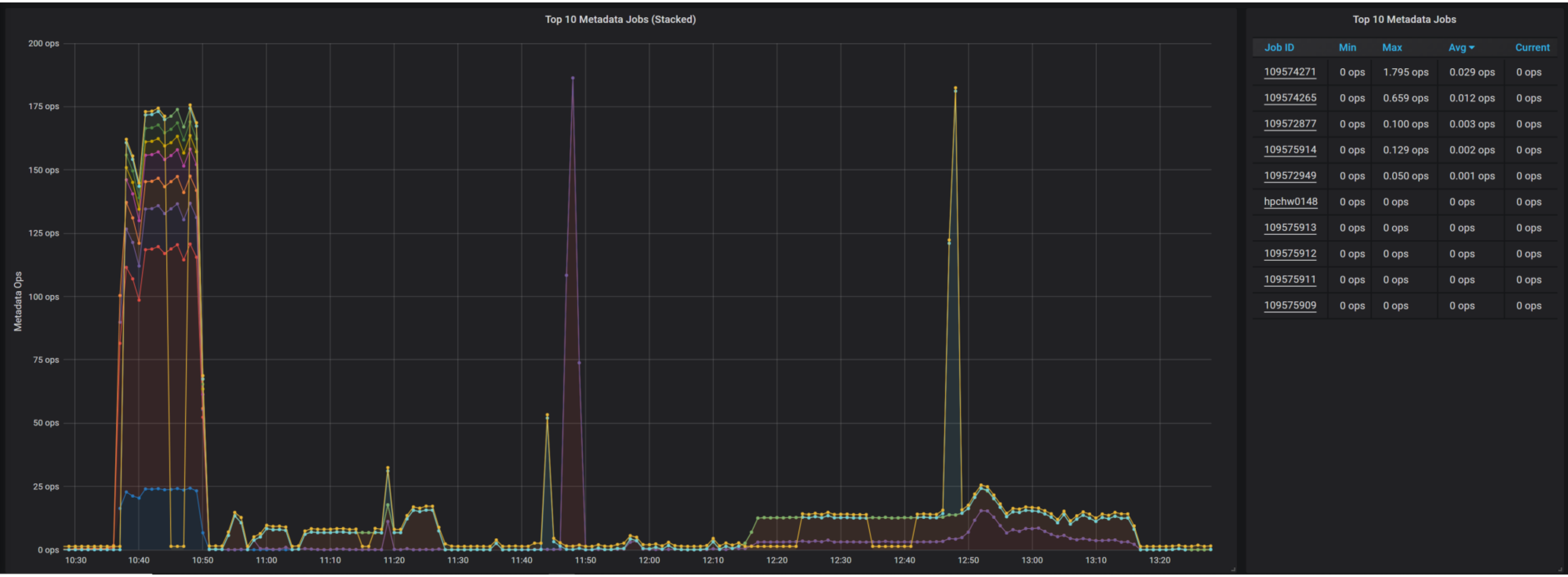
Lustre Top Jobs Dashboard



Top 10 Write Jobs

Job ID	Min	Max	Avg ▾	Current
109575914	0 Bs	376.790 MBs	260.015 MBs	0 Bs
hpchw0148	0 Bs	319.718 MBs	189.830 MBs	0 Bs
109575912	7.025 MBs	238.737 MBs	164.658 MBs	231.169 MBs
109575911	0 Bs	183.875 MBs	127.306 MBs	0 Bs
109575909	0 Bs	184.427 MBs	115.040 MBs	0 Bs
109575913	0 Bs	186.472 MBs	67.411 MBs	0 Bs
109574271	0 Bs	191.675 MBs	12.990 MBs	0 Bs
109572949	0 Bs	208.664 MBs	11.850 MBs	0 Bs
109574265	0 Bs	125.318 MBs	10.782 MBs	0 Bs
109572877	0 Bs	161.562 MBs	7.792 MBs	0 Bs

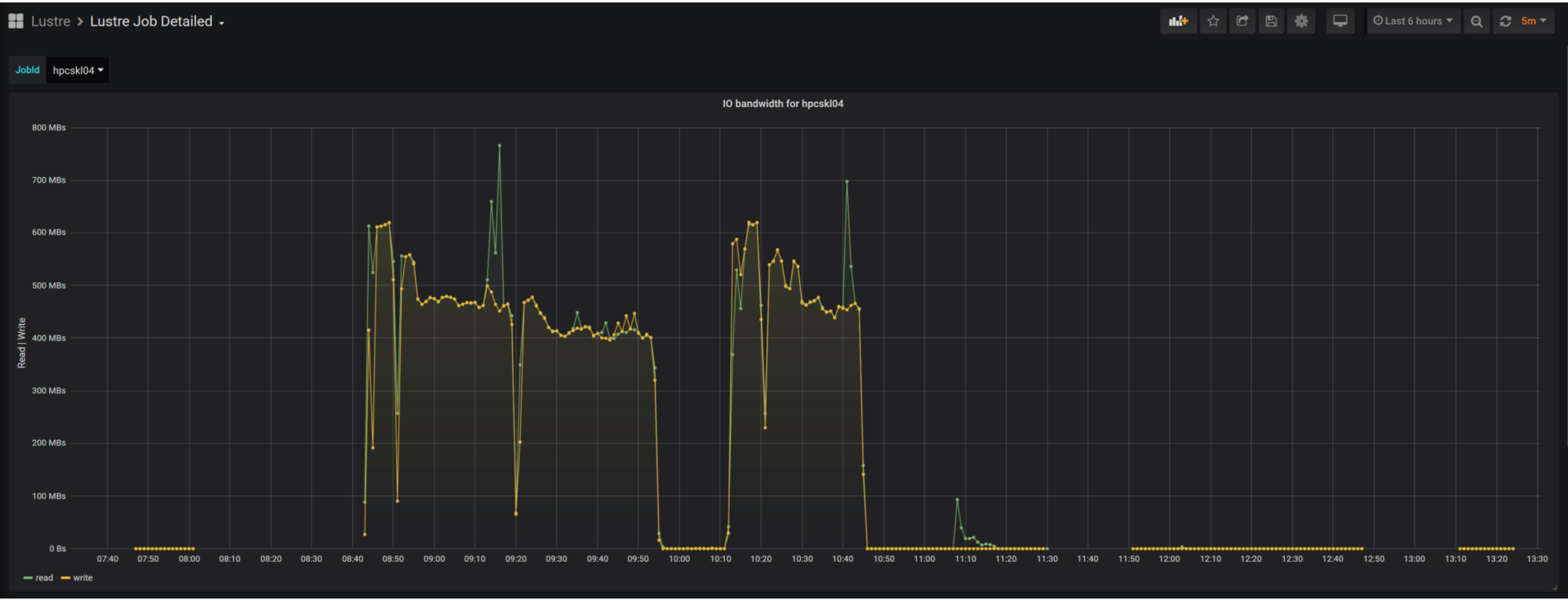
Lustre Top Jobs Dashboard



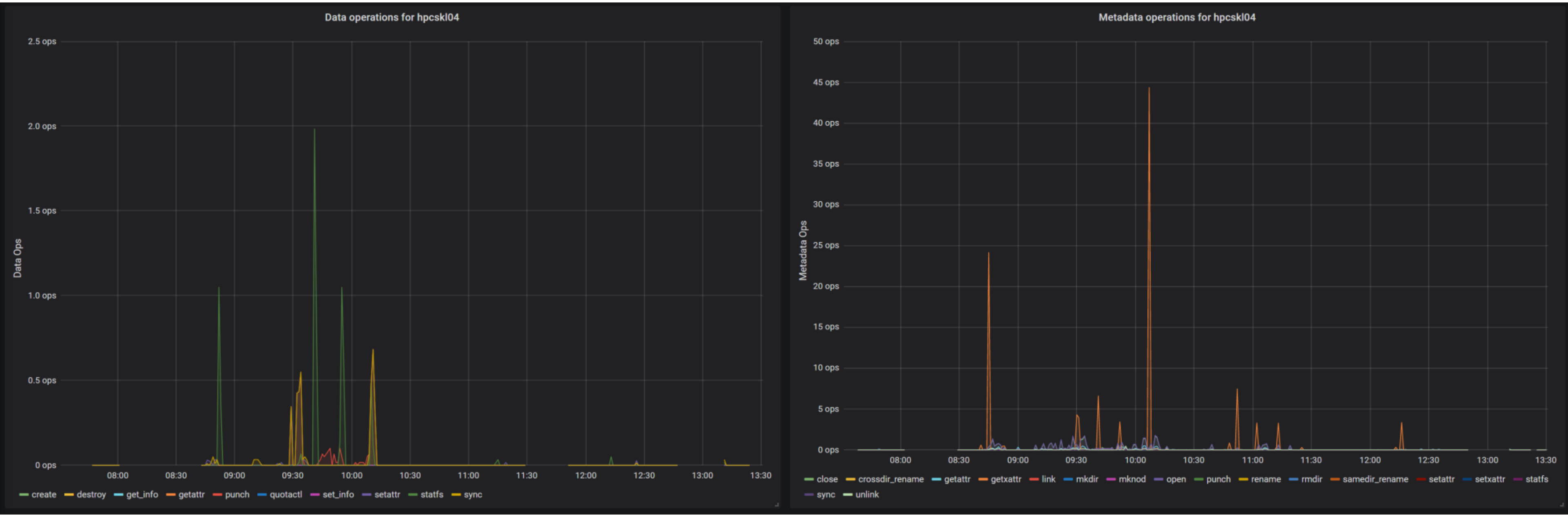
Top 10 Metadata Jobs

Job ID	Min	Max	Avg ▾	Current
109574271	0 ops	1.795 ops	0.029 ops	0 ops
109574265	0 ops	0.659 ops	0.012 ops	0 ops
109572877	0 ops	0.100 ops	0.003 ops	0 ops
109575914	0 ops	0.129 ops	0.002 ops	0 ops
109572949	0 ops	0.050 ops	0.001 ops	0 ops
hpchw0148	0 ops	0 ops	0 ops	0 ops
109575913	0 ops	0 ops	0 ops	0 ops
109575912	0 ops	0 ops	0 ops	0 ops
109575911	0 ops	0 ops	0 ops	0 ops
109575909	0 ops	0 ops	0 ops	0 ops

Lustre Job Detail Dashboard

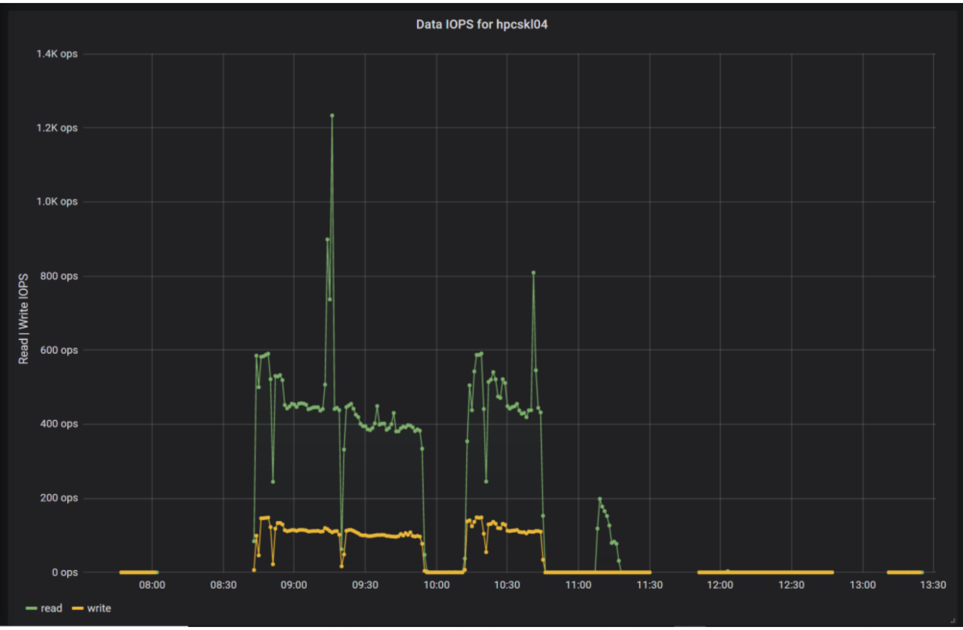


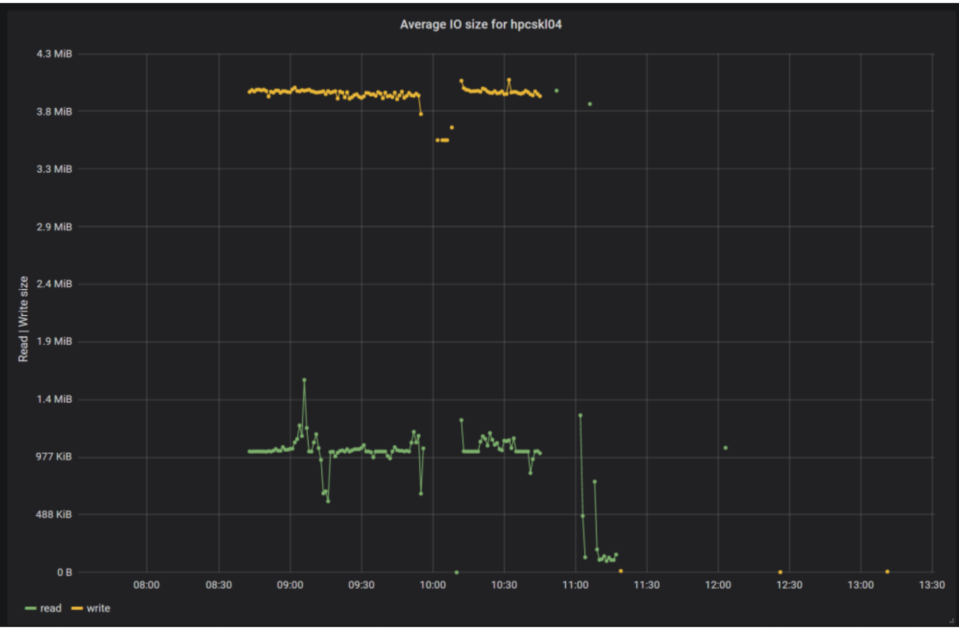
Lustre Job Detail Dashboard


Data operations for hpcskl04

Metadata operations for hpcskl04

Lustre Job Detail Dashboard





Word of warning

- Precompute what you want to visualize into new metric series to reduce burden on Prometheus servers when trying to respond to complex queries
- Everything in this software stack is healthy except the Lustre Exporter
- HPE is no longer going to support the Lustre Exporter
- Join us in supporting the open source Lustre Exporter

Questions?