



NORTH AMERICA | San Francisco 2024



**LET YOUR OPENSEARCH
CLUSTER MONITOR ITSELF**

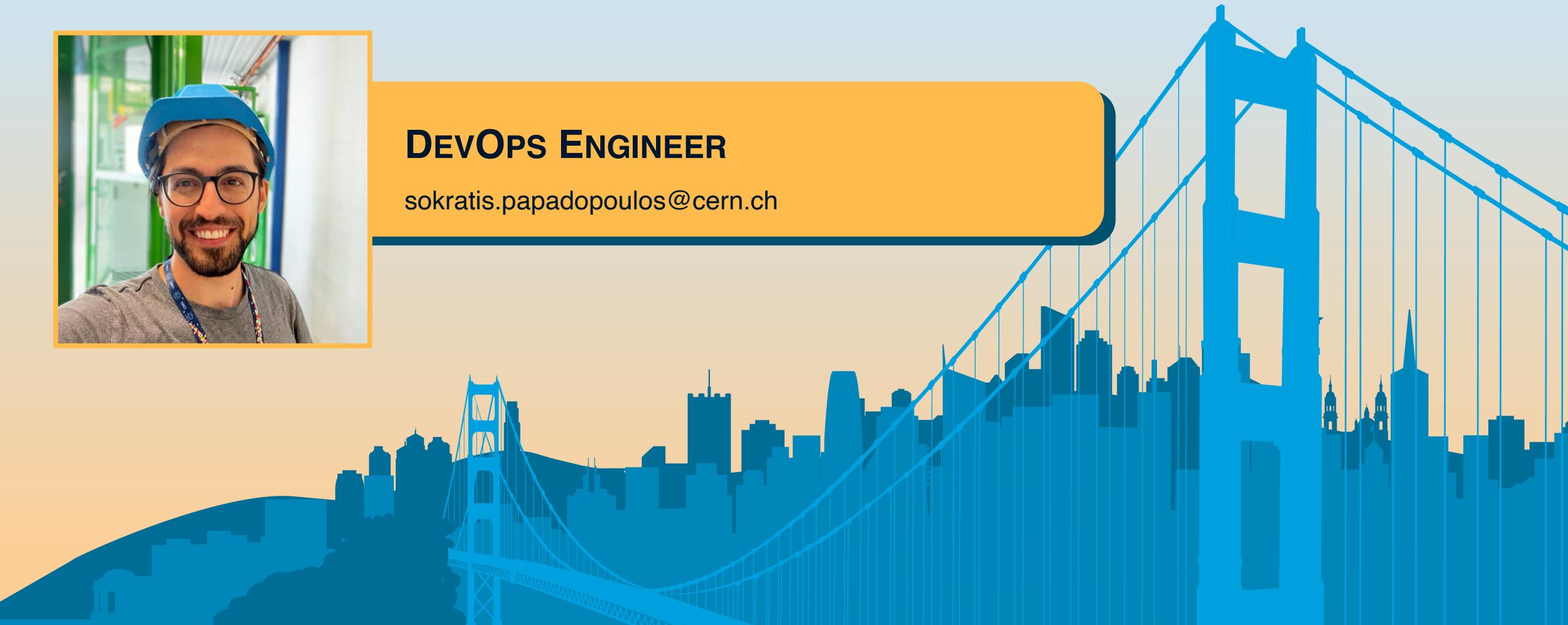
SOKRATIS PAPADOPoulos

OpenSearch service manager at CERN



DevOps ENGINEER

sokratis.papadopoulos@cern.ch



CERN

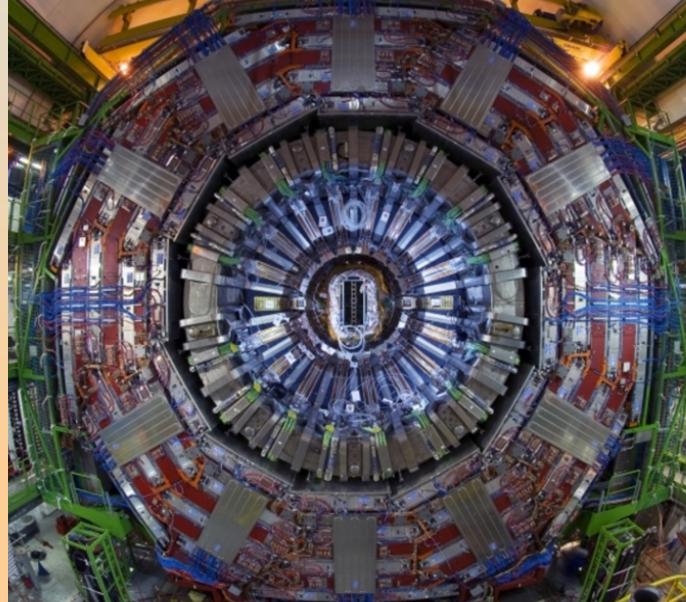
Uncover what the universe is made of & how it works

- 17,000 people from all over the world
- Particle accelerator facilities
- Push the frontiers of science and technology



WORLDWIDE LHC COMPUTING GRID

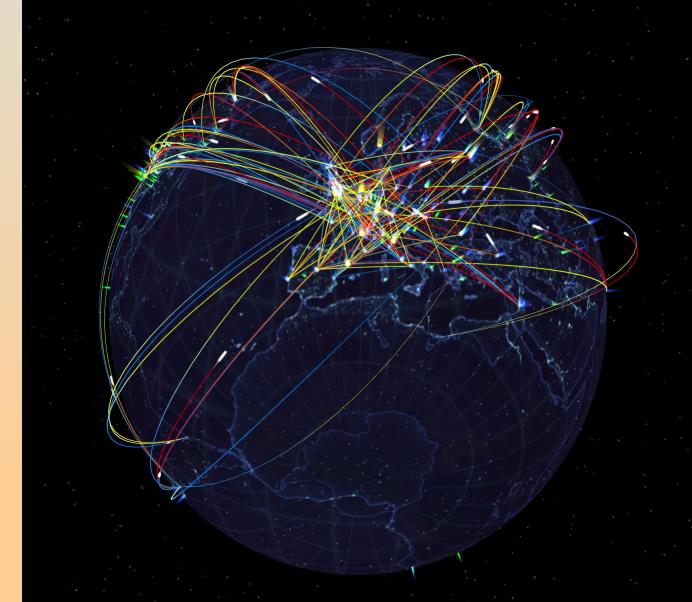
**Generating
1 PB per second**
only 1% is kept
(interesting events)



CERN data center
data reconstruction
tape archival
data distribution

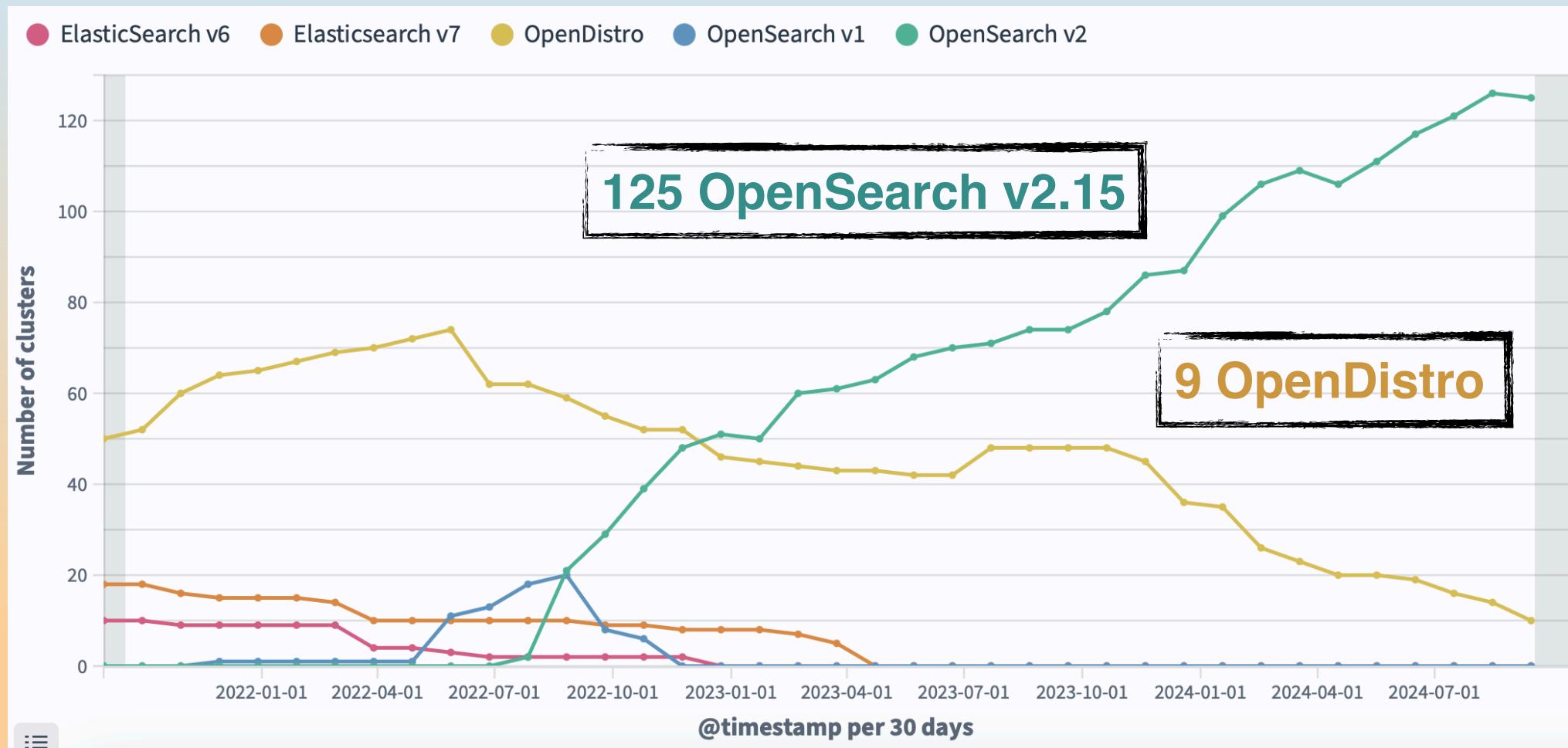


170 centers
36 countries
~200 PB of data/year
data analysis



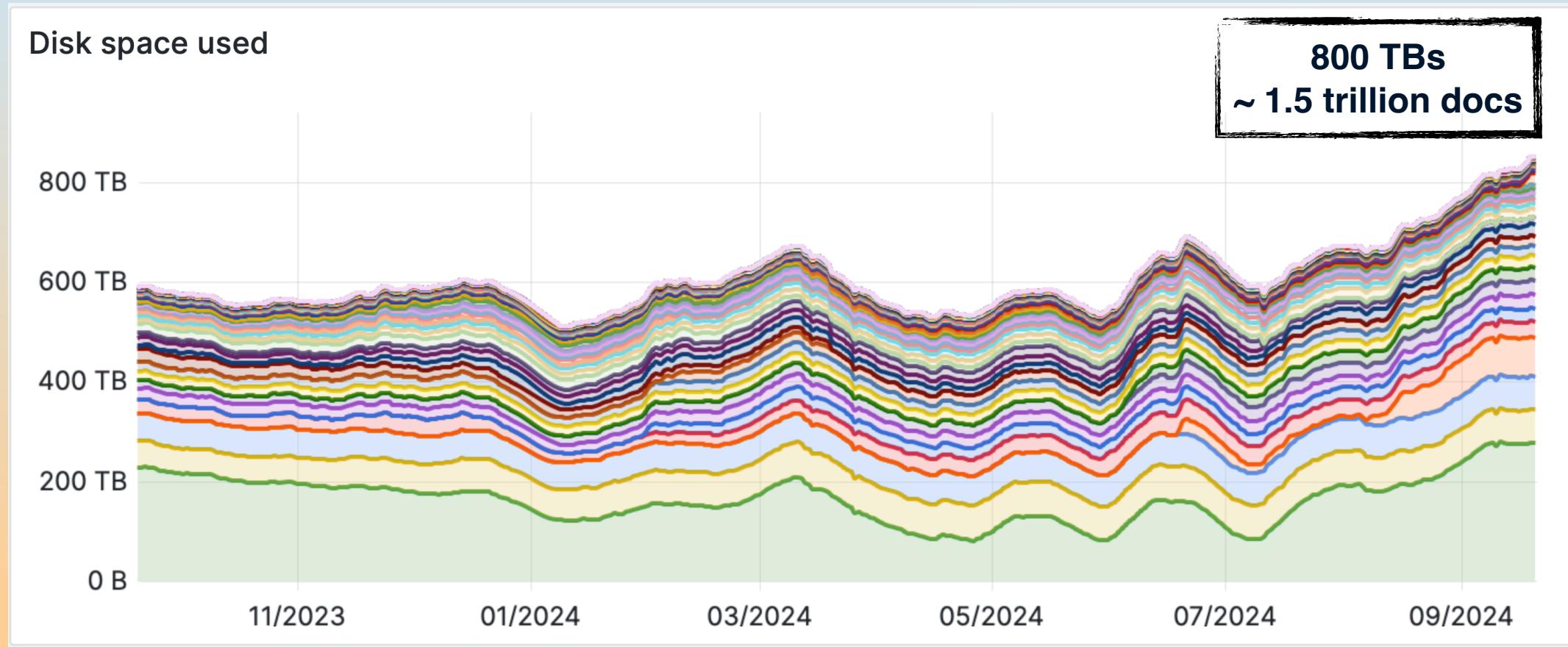
OPENSEARCH SERVICE AT CERN

From Elasticsearch to OpenDistro to OpenSearch



OPENSEARCH SERVICE AT CERN

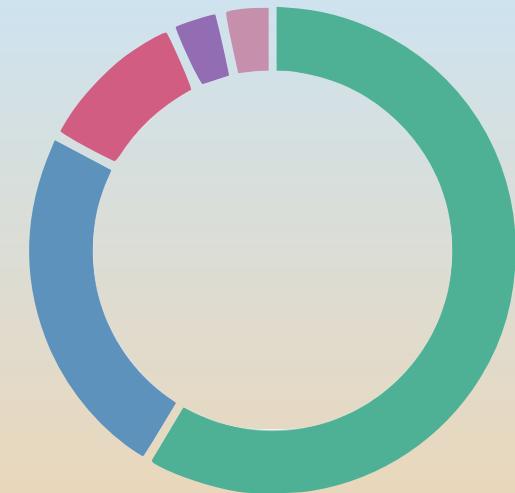
Used disk space evolution over the past 3 years



OPENSEARCH SERVICE AT CERN

Who is using it?

- Experiments
 - ATLAS, CMS, ALICE, LHCb, etc.
- HEP community search engines
 - [Zenodo](#) & [Inspire](#)
- IT
 - Security, Monitoring, Storage, etc.
- WLCG monitoring



- Log/event analytics
- Full-text search
- Other
- Geospatial data an...
- Machine Learning

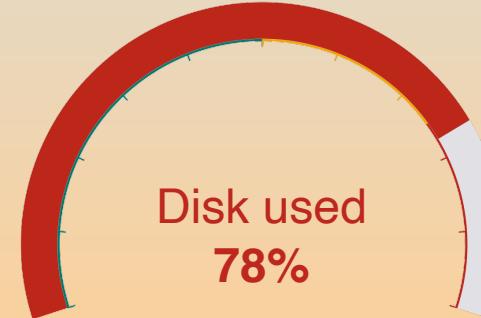
WHAT IS THE CHALLENGE?

Addressing common poor practices in OpenSearch usage

- Growing number of OpenSearch clusters
- Observed common bad practices on OpenSearch usage
 - Leading to high resource utilization & poor performance



Too big shards



Node getting full

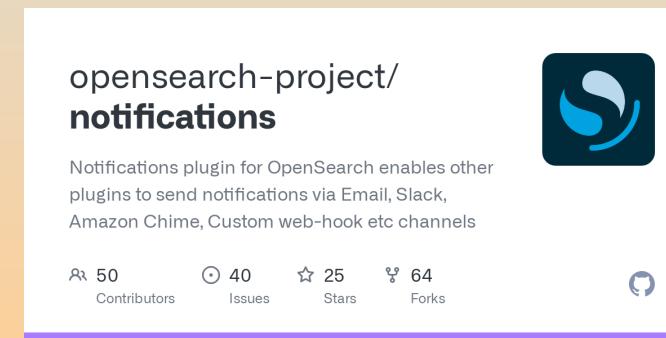
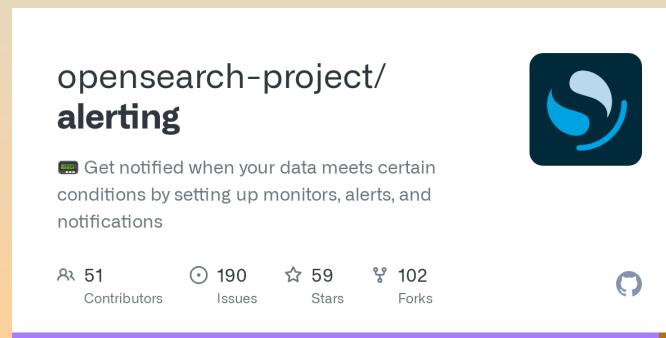


Too many shards

WHAT IS THE CHALLENGE?

Addressing common poor practices in OpenSearch usage

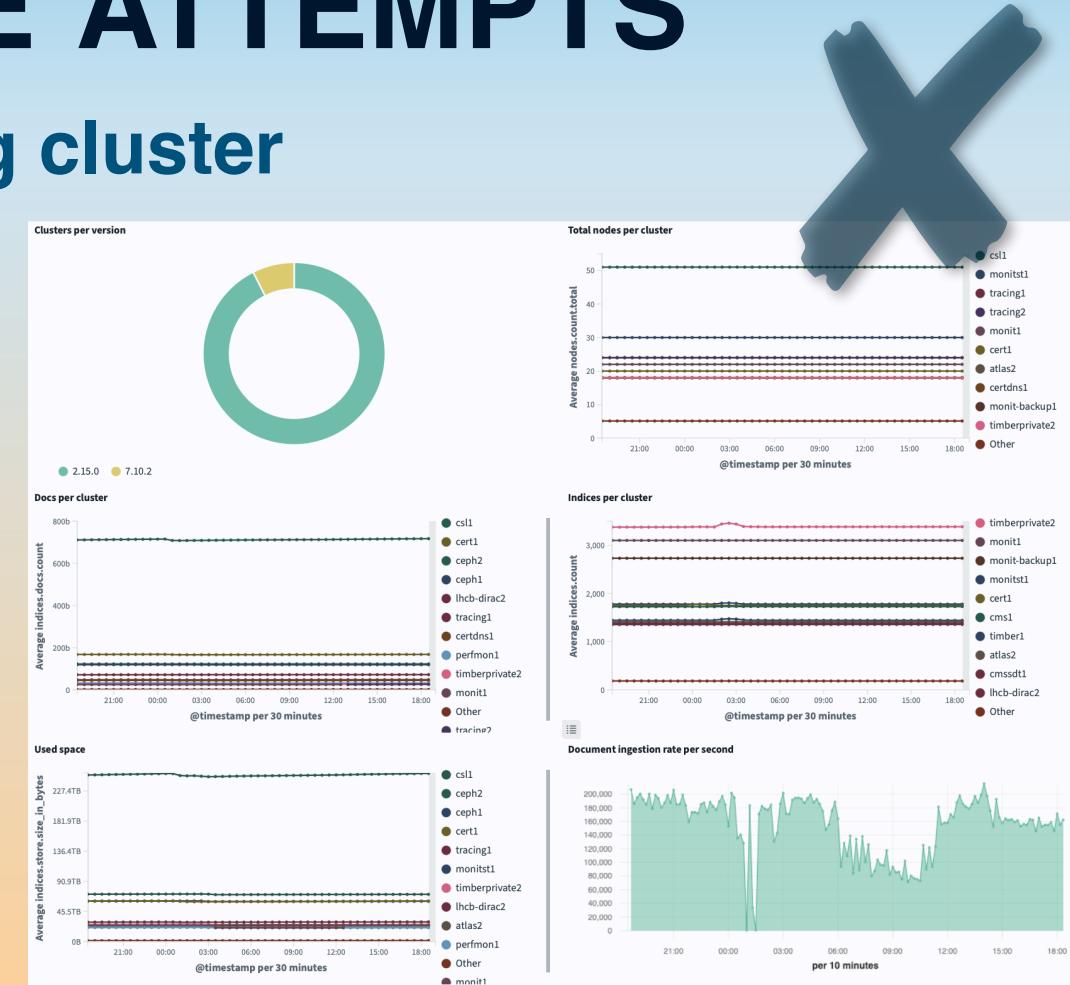
- Part of the responsibility falls on our customers' shoulders
 - Each one has a different communication channel
- **Alerting & Notifications** plugins come to the rescue!



MONITORING USAGE ATTEMPTS

Attempt 1: Centralized monitoring cluster

- Query customers' cluster APIs
 - Using e.g., Logstash
- Store responses in central cluster
- Create **Monitors** on top of those
- 100s of **Notification channels**
- One *trigger* for each cluster
 - limited to 10 triggers



MONITORING USAGE ATTEMPTS

Attempt 2: Decentralized monitoring

- Create **default notification channels** upon cluster bootstrap
 - Slack, Email, Webhook, etc.

Name
Mattermost channel for cluster admins

Notification status
● Active

Type
Slack

Description
MM channel with all cluster admins used for communication and cluster/data alerts

Name
SNOW

Notification status
● Active

Type
Custom webhook

Description
Send alerts to your FE in ServiceNow as a GNI ticket

Name
Email channel for cluster admins

Notification status
● Active

Type
Email

Description
Send an email to cluster admins egroup

MONITORING USAGE ATTEMPTS

Attempt 2: Decentralized monitoring

- Create **default monitors** upon cluster bootstrap
 - Analyzing all common OpenSearch usage problems
 - Targeting respective notification channels



Node stats

Cluster health

Shards number

Snapshots

Disk watermarks

Zero replicas

Tasks

Cluster managers

Transient settings

Shards size

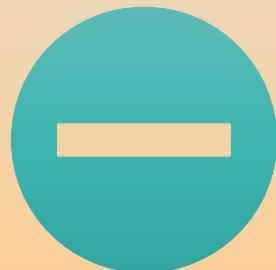
Indexes size

DECENTRALISED MONITORING

Pros and cons



- No single point of failure
- Better scalability
- No need to store metrics
- No external tools/dependencies



- Maintain monitoring objects across clusters
- Does not work if cluster is down

DECENTRALISED MONITORING

Implementation: Scripting

- Clusters config is stored in yaml files on GitLab
 - Customer channels
 - Allocated resources
- Created 12 monitor templates
 - Placeholders for cluster-specific values
 - Placeholders for notification channel
- Using *opensearch-py* library to create objects



{...} cluster_health.json
{...} cluster_health_yellow.json
{...} cluster_managers.json
{...} disk_watermarks.json
{...} indexes_size.json
{...} node_stats.json
{...} shards_number.json
{...} shards_size.json
{...} snapshots.json
{...} tasks.json
{...} transient_settings.json
{...} zero_replicas.json

DECENTRALISED MONITORING

Implementation: Monitor configuration

- Using **Per cluster metrics** monitors
- Periodic API calls, e.g.,
 - GET _cluster/health
 - GET _cluster/stats
 - GET _node/stats
- Set trigger conditions based on responses
- Notify the appropriate channel

Request type
Specify a request type to monitor cluster metrics such as health, JVM, and CPU usage. [Learn more](#)

Cluster health

Path parameters - optional
Filter responses by providing path parameters for the cluster health API. [Learn more](#)

GET /_cluster/health/ e.g., indexAlias1,indexAlias2...

Preview query

Response

```
1 { "number_of_pending_tasks": 0,
2   "cluster_name": "playground1",
3   "active_shards": 1057,
4   "active_primary_shards": 527,
5   "unassigned_shards": 0,
6   "delayed_unassigned_shards": 0,
7   "timed_out": false,
8   "discovered_cluster_manager": true,
9   "discovered_master": true,
10  "relocating_shards": 0,
11  "initializing_shards": 0,
12  "task_max_waiting_in_queue_millis": 0,
13  "number_of_data_nodes": 6,
14  "number_of_in_flight_fetch": 0,
15  "active_shards_percent_as_number": 100,
16  "status": "green",
17  "number_of_nodes": 12 }
```

DECENTRALISED MONITORING

Implementation: Monitor configuration

```
{  
  "name": "Shards number",  
  "type": "monitor",  
  "monitor_type": "cluster_metrics_monitor",  
  ...  
  "inputs": [  
    {  
      "uri": {  
        "api_type": "CLUSTER_HEALTH",  
        "path": "_cluster/health",  
        ...  
      }  
    }  
  ]  
}
```

```
...  
  "subject_template": {  
    "source": "High number of shards in cluster  
    ** __cluster_name__ **"  
  }  
}
```

```
"actions": [  
  {  
    "name": "Notify Mattermost channel",  
    "destination_id": "__customers_channel_id__",  
    "message_template": { ...  
  
    - shards per data node:  
      **{{ctx.results[0].shards_per_node}}**  
      (threshold: __shards_number_threshold__)  
  
    Check your [index policies]  
    (https://\_\_cluster\_alias\_\_.cern.ch/.../index-policies)  
    ...  
  }]
```

DECENTRALISED MONITORING

OpenSearch Dashboards UI: Notification channels

Channels (4)

<input type="checkbox"/> Name ↑	Notification status	Type	Description
<input type="checkbox"/> Email channel for cluster admins	● Active	Email	Send an email to cluster admins egroup
<input type="checkbox"/> Mattermost channel for OpenSearch team	● Active	Slack	MM channel internal to the OpenSearch team used for infrastructure alerts
<input type="checkbox"/> Mattermost channel for cluster admins	● Active	Slack	MM channel with all cluster admins used for communication and cluster/data alerts
<input type="checkbox"/> SNOW	● Active	Custom webhook	Send alerts to your FE in ServiceNow as a GNI ticket

Rows per page: 10 ▾ < 1 >

DECENTRALISED MONITORING

OpenSearch Dashboards UI: Monitors

Monitors								
<input type="text"/> Search								
<input type="checkbox"/>	Monitor name ↑	State	Type	Latest alert	Last notification time	Active	Acknowledged	Errors
<input type="checkbox"/>	Cluster health	Enabled	Per cluster metrics	--	-	0	0	0
<input type="checkbox"/>	Cluster health yellow	Enabled	Per cluster metrics	--	-	0	0	0
<input type="checkbox"/>	Cluster managers	Enabled	Per cluster metrics	--	-	0	0	0
<input type="checkbox"/>	Disk watermarks	Enabled	Per cluster metrics	--	-	0	0	0
<input type="checkbox"/>	Indexes size	Enabled	Per cluster metrics	zero docs	09/02/24 2:00 pm	0	0	0
<input type="checkbox"/>	Node stats	Enabled	Per cluster metrics	--	-	0	0	0
<input type="checkbox"/>	Shards number	Enabled	Per cluster metrics	--	-	0	0	0
<input type="checkbox"/>	Shards size	Enabled	Per cluster metrics	--	-	0	0	0
<input type="checkbox"/>	Snapshots	Enabled	Per cluster metrics	--	-	0	0	0
<input type="checkbox"/>	Tasks	Enabled	Per cluster metrics	--	-	0	0	0
<input type="checkbox"/>	Transient settings	Enabled	Per cluster metrics	--	-	0	0	0
<input type="checkbox"/>	Zero replicas	Enabled	Per cluster metrics	zero replicas	08/12/24 8:29 am	0	0	0

ALERTING EXAMPLES

High number of shards in a node



opensearch BOT 14:00

High number of shards in cluster zenodo-prod1

Each index consists of shards, typically 1 + 1 replica and they capture space in memory. The cluster is currently configured to handle less shards than it actually does and this may lead to performance issues.

- shards: **1192**
- data nodes: **3**
- shards per data node: **397** (threshold: 240)

Please ensure your [index policies](#) are set up properly. Also, often times you have a high number of small shards, which you can reduce by moving from daily indices (YYYY-MM-DD) to monthly ones (YYYY-MM), as explained [here](#).

If in doubt, please contact the OpenSearch service managers for assistance.

ALERTING EXAMPLES

Too big shards



opensearch BOT 14:00

Big shards found on perfmon1

Big shards can lead to several performance and operational issues, including cluster imbalance. Please consider increasing the number of shards allocated on respective indexes, by modifying the relevant [index template](#).

- `perfmon_logstash-audit-authenticated-2024.07.17` storing `367012719` docs with total size `125.3` GB on node `oscperfmon102-perfmon1_data3`

ALERTING EXAMPLES

Zero replicas



opensearch BOT 17:59

⚠ Zero replicas found on some `tiny1` indexes ⚠

Setting zero replicas for an index is highly discouraged as it makes your data vulnerable to permanent loss in the event of a node failure. In production environments, it's essential to have replicas for data redundancy and high availability.

In order to fix it, run the following command from your [devtools console](#):

```
PUT index_name/_settings
{
  "settings": {
    "number_of_replicas": 1
  }
}
```

Then, ensure that you do not have any [index template](#) that sets replicas to "0" for new indexes.

ALERTING EXAMPLES

Index with zero docs



opensearch BOT 14:00

Indexes with **zero docs** found in **cds-ils1**

This is a bad practice. Please **take one of the following actions**:

- if they are no longer useful, simply delete them by running the following command from your [devtools console](#):

```
DELETE eitems-eitem-v1.0.0
DELETE eitems-eitem-v2.0.0-1718273496
DELETE items-item-v1.0.0-1678463505
DELETE series-series-v1.0.0-1678463505
```

- if they are still used, but have a low document rate, convert your index to rotate on monthly or yearly basis.

If in doubt, please contact the OpenSearch service managers for assistance.

ALERTING EXAMPLES

Cluster health



centralised_monitors BOT 12:10

⚠️ Clusters with **yellow** status found ⚠️

- [tim1](#): status is **yellow** for more than 30 minutes

For more info, click [here](#) or consult the [rota procedures](#).



centralised_monitors BOT 12:15

❗️ Clusters with **red** status found ❗️

- [tim1](#): status is **red** for more than 30 minutes

For more info, click [here](#) or consult the [rota procedures](#).

SUMMARY

Stay on the optimal path, or get alerted!

- The decentralised monitor solution scales freely
- External monitoring is needed for cluster health
- Customers act upon informative alerts on the right channel
- Customers can trust that any missteps will trigger timely alerts

