IT Monitoring Overview

Nikolay Tsvetkov

4 December 2023

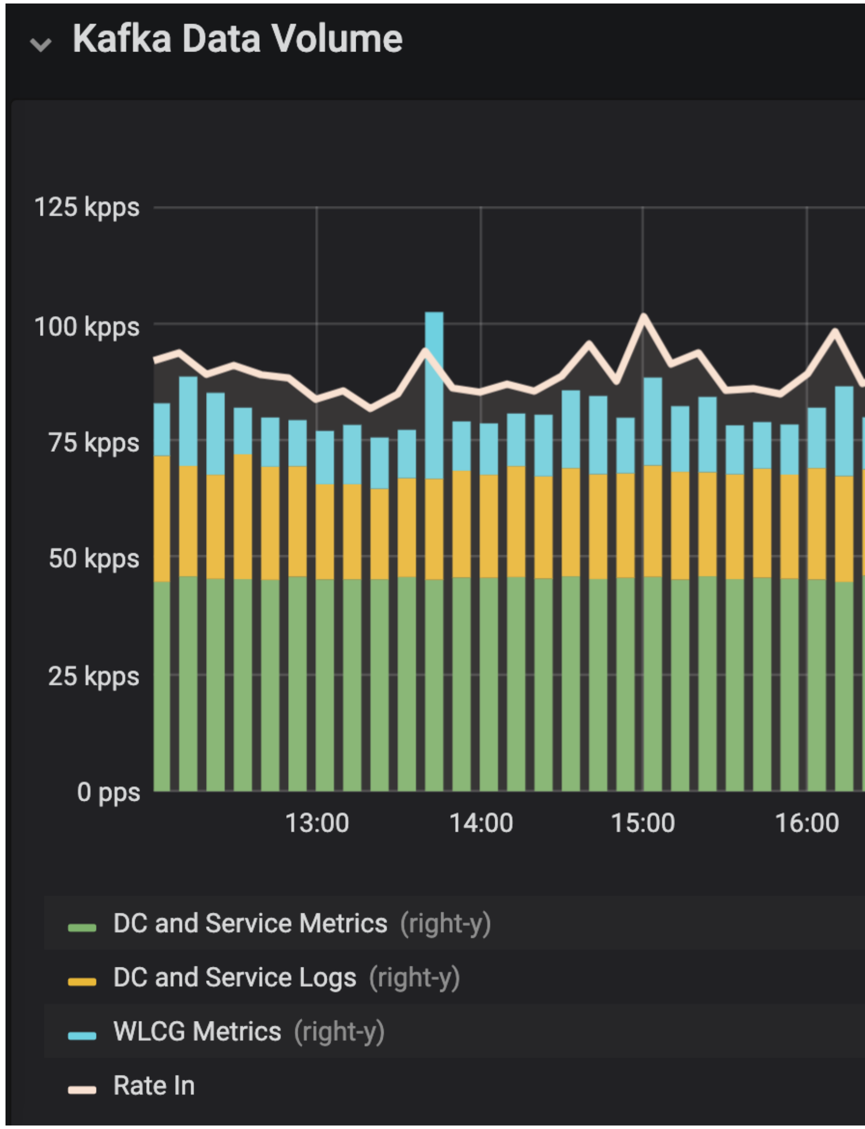
IT Monitoring Service (MONIT)

Central monitoring service for IT, CERN Data Center and the WLCG collaboration

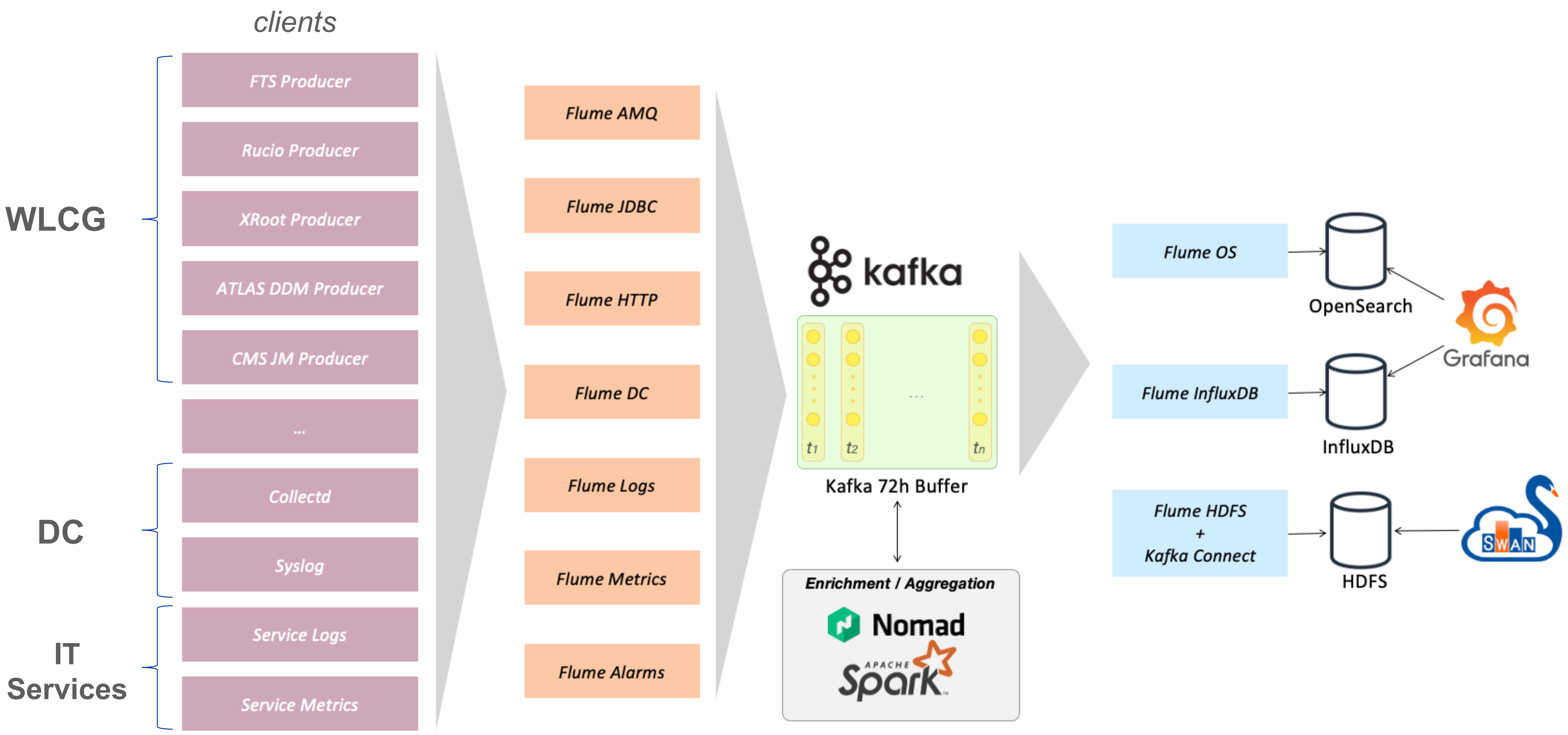
Stores and processes *metrics* and *logs* from applications and infrastructure

MONIT in numbers

- **Data rate**:
  - ~ 85k documents/s
  - ~ 3.5 TB/day (compressed)
- **Data volume**: ~ 500 TB (compressed)
- **Grafana**: ~ 5000 users



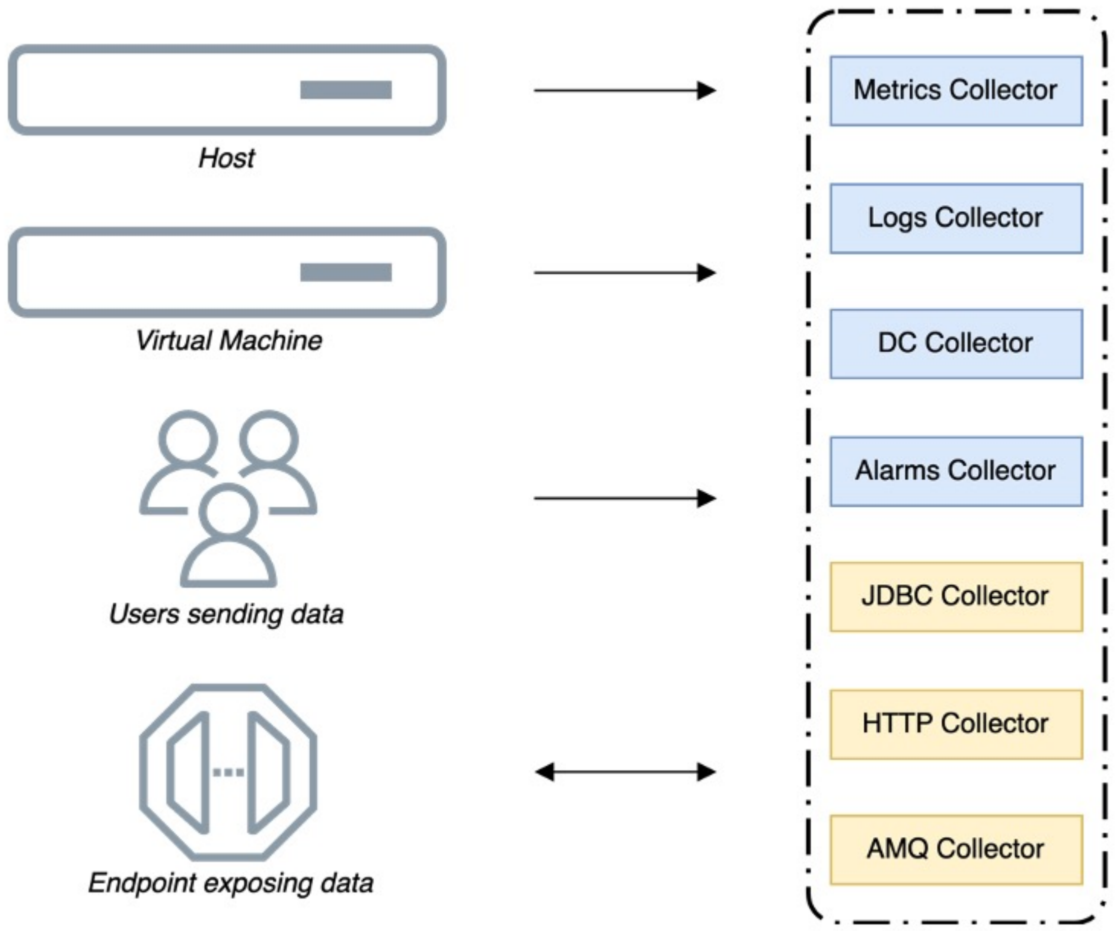
Architecture



Ingestion Layer

Based on **Apache Flume**

- Internal channel for data buffering (disk or memory)
- Data validation and transformation using *Flume Interceptors* and *Kite Morphlines*
- Accepts JSON documents with required metadata fields



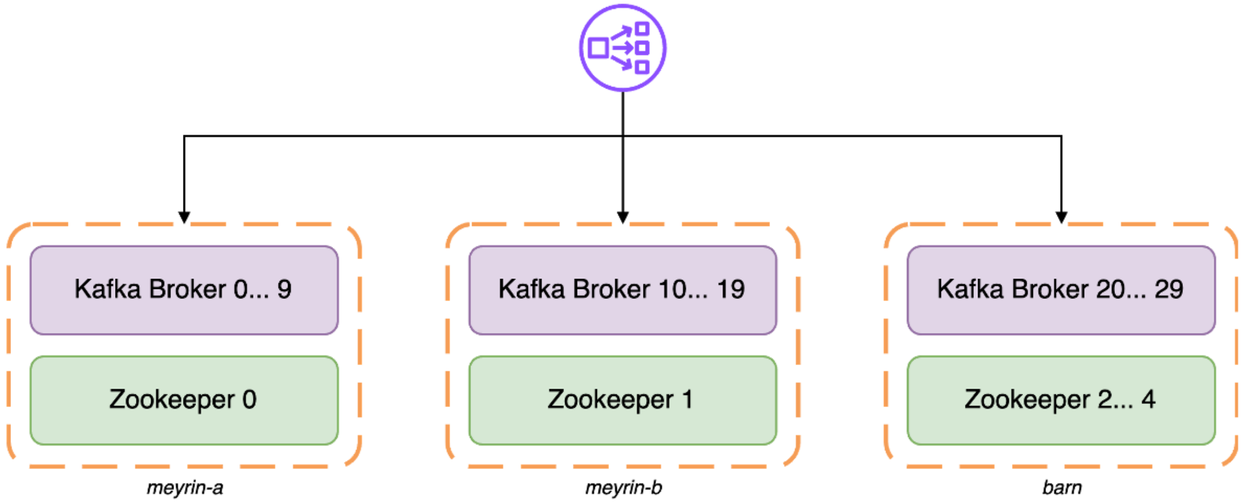
Deployment details

- Standalone agents per data source types
  - HTTP, Avro RPC (push)
  - AMQ, JDBC, HTTP (pull)
- Behind DNS load balancer and HAProxy in some cases
  - Scales horizontally
- 65 agent instances in total (VMs in 3 AVZs)

Transport Layer (Kafka)

Kafka v3.4 cluster provided by the **Streaming Service (NILE)**

- 30 brokers split in 3 AVZs (one of the zones in the BARN)
  - m2.xLarge nodes (VMs)
  - CEPH volumes split in 3 AVZs (respect the brokers split)
- 5 Zookeeper nodes in 3 AVZs (3 nodes in the BARN)
- Topic ACLs by users/egroups



Configuration and topic management

- Topic per producer and data "type"
  - 30 topic partitions with 3 replicas
  - Total: ~ 300 topics / 27k partitions (with replica)
- Round-robin partitioning strategy
- 3 days retention period

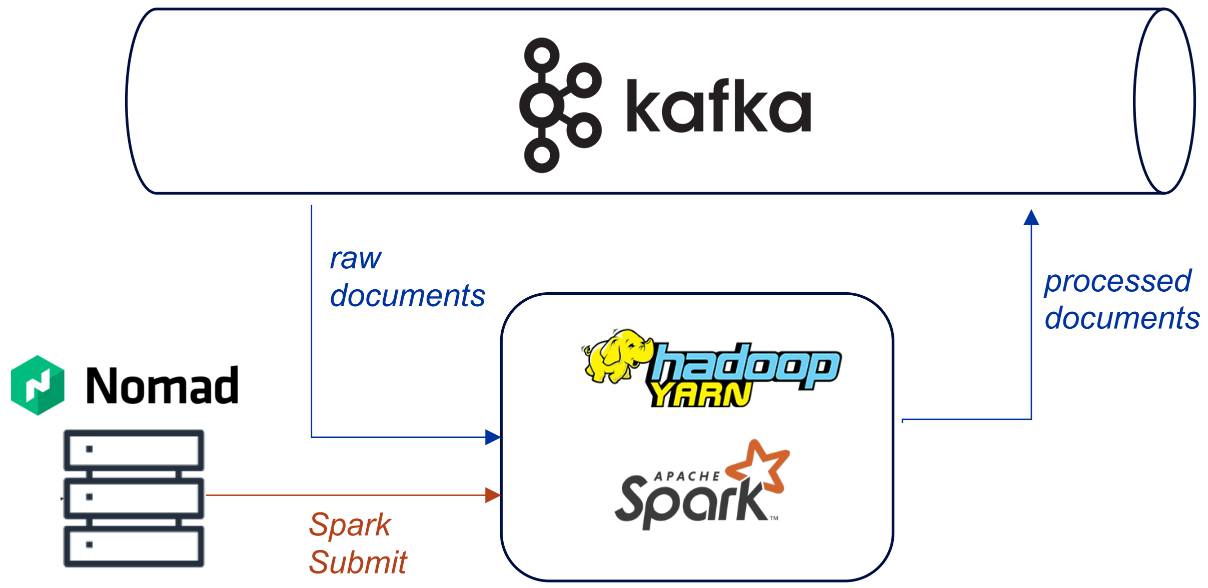
Topic naming schema: <producer>_<type_prefix*>[_<type>] (e.g. fts_raw_complete, fts_enr_complete)

*type_prefix – represents fixed set of documet types (**raw, agg, enr, logs**)

Processing Layer

Spark jobs for data aggregation & enrichment

- Running *Spark v3.4*
- Managed by Nomad (streaming and batch)
- Submitted in **YARN** cluster mode on **Analytix**
  - 17 production jobs
  - Quota: 1200 vCores/ 7TB RAM
  - Using queues to manage resource allocation
- DEV jobs submitted on **HadoopQA**



Deployment managed through GitLab CI

- Docker image built per job
- Nomad job definition per environment/cluster

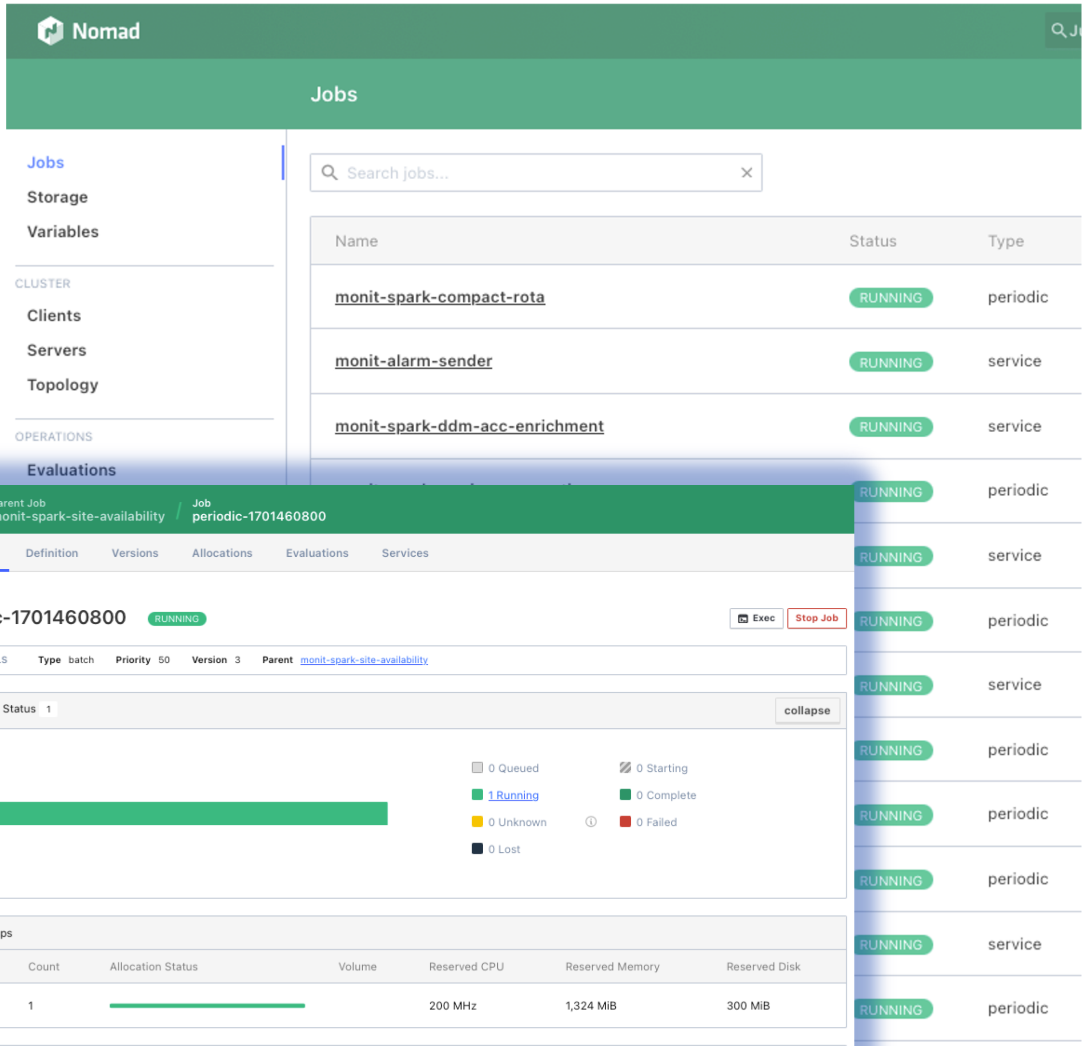
Processing Layer (Nomad)

Nomad by HashiCorp

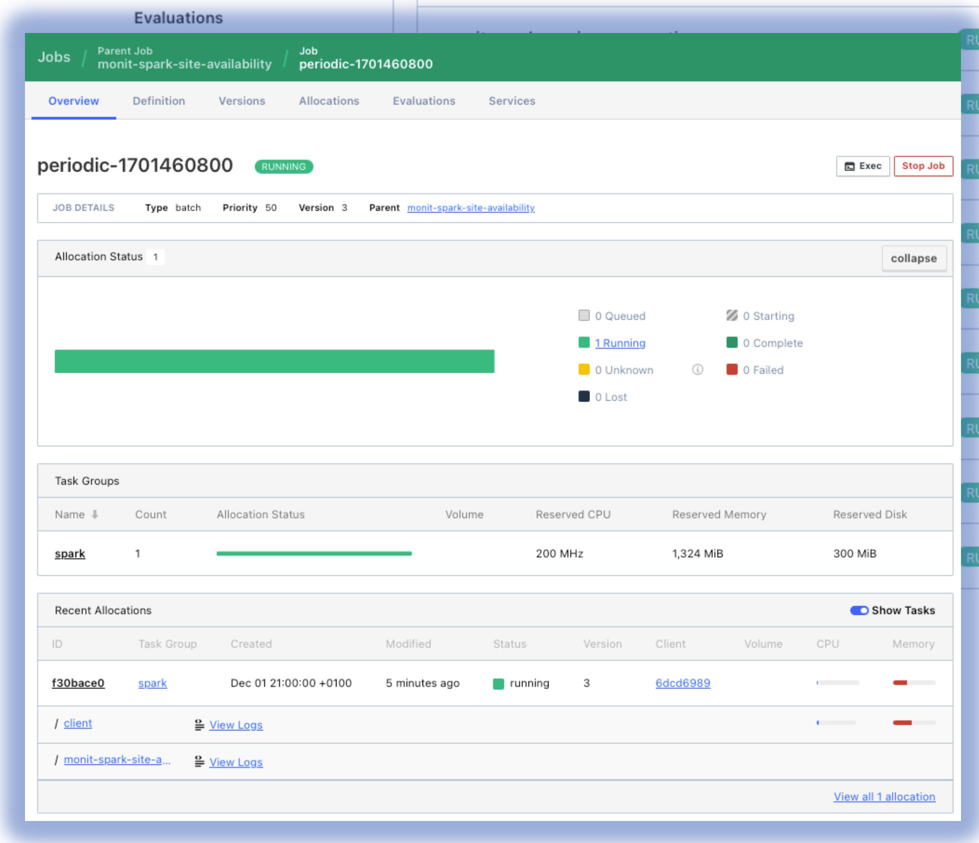
- Job scheduler and orchestrator
- Manages containers and non-containerized applications
- Free for “own use” on-premise deployment
- Scales horizontally (add more nodes)

Nomad in MONIT

- Using 6 “client” nodes on a shared IT-DA cluster
- Running Spark streaming as “service” and batch as “periodic” Nomad jobs
- Little resources required in Nomad (Spark driver running in *cluster* mode)



Name	Status	Type
monit-spark-compact-rota	RUNNING	periodic
monit-alarm-sender	RUNNING	service
monit-spark-ddm-acc-enrichment	RUNNING	service
	RUNNING	periodic
	RUNNING	service
	RUNNING	periodic
	RUNNING	service
	RUNNING	periodic
	RUNNING	periodic
	RUNNING	periodic
	RUNNING	service
	RUNNING	periodic



periodic-1701460800 RUNNING

JOB DETAILS Type batch Priority 50 Version 3 Parent monit-spark-site-availability

Allocation Status 1

0 Queued	0 Starting
1 Running	0 Complete
0 Unknown	0 Failed
0 Lost	

Task Groups

Name	Count	Allocation Status	Volume	Reserved CPU	Reserved Memory	Reserved Disk
spark	1			200 MHz	1,324 MiB	300 MiB

Recent Allocations

ID	Task Group	Created	Modified	Status	Version	Client	Volume	CPU	Memory
f30bace0	spark	Dec 01 21:00:00 +0100	5 minutes ago	running	3	6dcd6989			
/ client	View Logs								
/ monit-spark-site-a...	View Logs								

View all 1 allocation

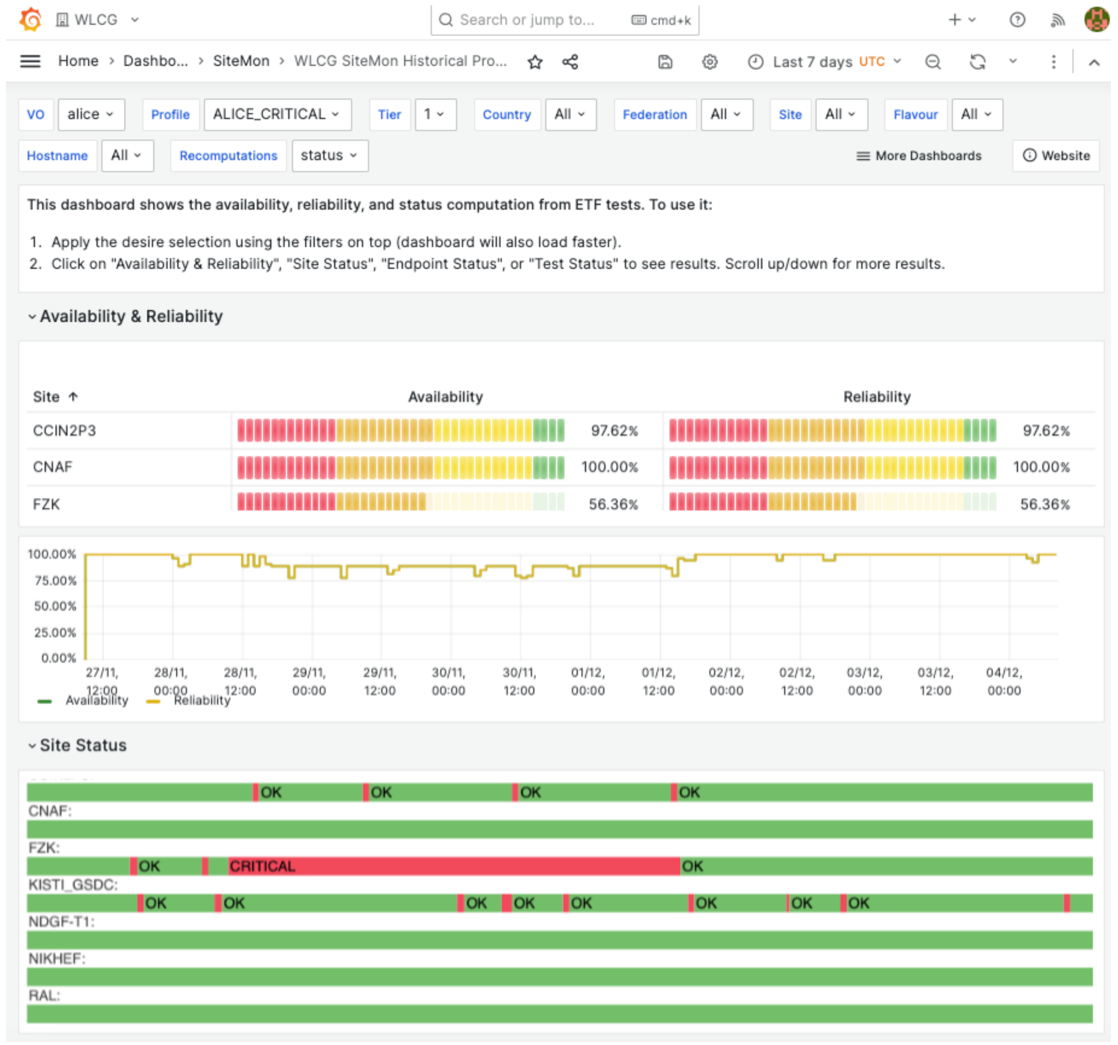
Spark jobs

- **Document enrichment**
  - Adding extra fields with information received from other sources (e.g. WLCG Topology info)
  - Join data from different Kafka topics
- **Data aggregation**
  - Downsampling data by aggregating for set of fields and longer intervals (e.g. 1h)
  - Usually applies on already enriched documents
- **Data recovery from HDFS**
  - Recover data for predefined interval directly from HDFS
  - In case of specific interval or data not available in Kafka anymore
- **Data compaction in HDFS**
  - Deduplicating and compacting files for past days
  - Deleting “too old” data after predefined retention period

Spark jobs (WLCG SiteMon)

WLCG Site Monitoring

- Calculates the availability/reliability of WLCG sites
- Combines data from different sources
- Dynamically configured through site “profiles”
  - Through GitLab repository
- Creates status result per 10 minutes interval
- Handle “site status” in case of missing data
  - Applies previous status for configured “TTL” interval
  - Sets UNKNWON after the “repeat” interval


WLCG
Search or jump to...
cmd+k
Home › Dashbo... › SiteMon › WLCG SiteMon Historical Pro...
Last 7 days UTC
More Dashboards
Website
Availability & Reliability
Site Status

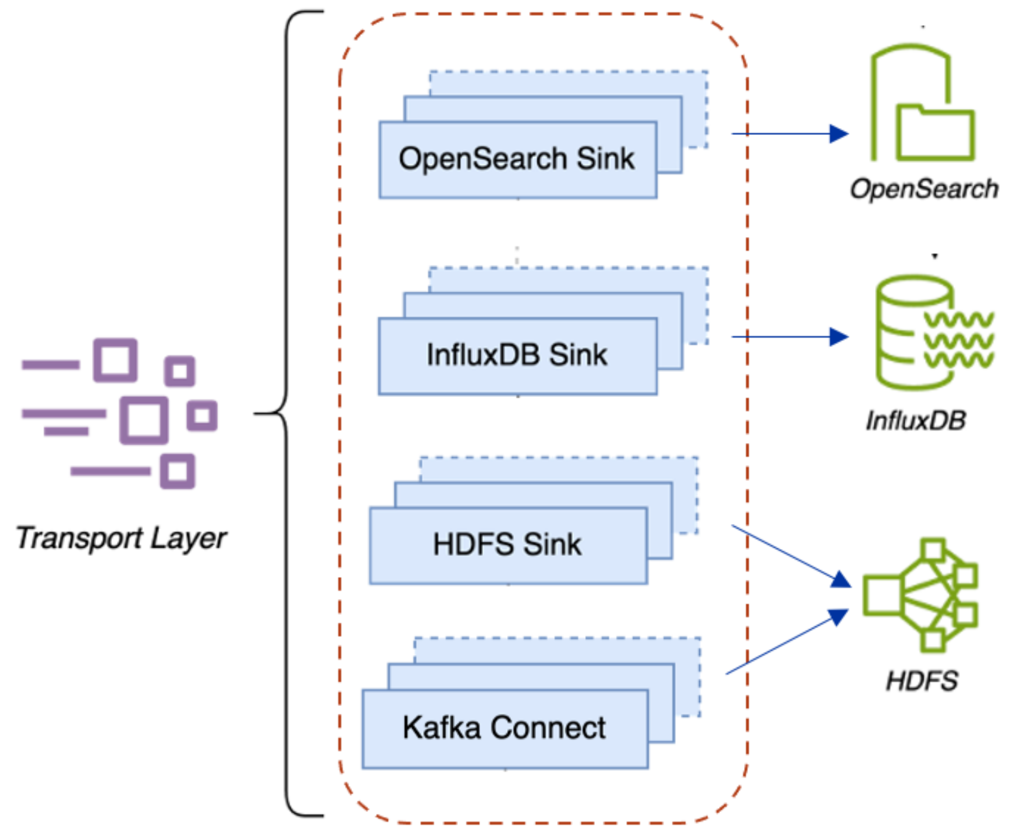
VO: alice | Profile: ALICE_CRITICAL | Tier: 1 | Country: All | Federation: All | Site: All | Flavour: All | Hostname: All | Recomputations: status

This dashboard shows the availability, reliability, and status computation from ETF tests. To use it:

1. Apply the desire selection using the filters on top (dashboard will also load faster).
2. Click on "Availability & Reliability", "Site Status", "Endpoint Status", or "Test Status" to see results. Scroll up/down for more results.

Site ↑	Availability	Reliability
CCIN2P3	97.62%	97.62%
CNAF	100.00%	100.00%
FZK	56.36%	56.36%

Sink Layer



Apache Flume as the main sink agent

- Writing to InfluxDB, OpenSearch and **HDFS**
- 60 instances in total (VMs in 3 AVZs)
- Kafka consumer group shared across instances of same type

Kafka Connect

- Writing Collectd data to **HDFS**
  - Using Confluent HDFS connector
- Connect cluster provided by NILE
  - 15 workers (VMs in 3 AVZs)
  - 50 connector instances
- Avro schema
  - Using static schema configuration

HDFS Storage

- **Analytix** cluster (production data)
  - ~4.2 million objects in directories / 5 million quota
  - ~1.10PB storage (with replication) / 1.5PB quota
- **HadoopQA** cluster for the QA infrastructure
- **Time based partitioning**
  - **/project/monitoring/**producer/type_prefix[/type]/year/month/day
  - *Parquet* with *Snappy* compression or *JSON.GZ* files depending of the flow
  - Compaction job runnign daily (*Spark*)
- **Enforced retention policy as per OC11**
  - Exceptions apply in case of approval from the Data Privacy Office
- **ACLs management for private folders**

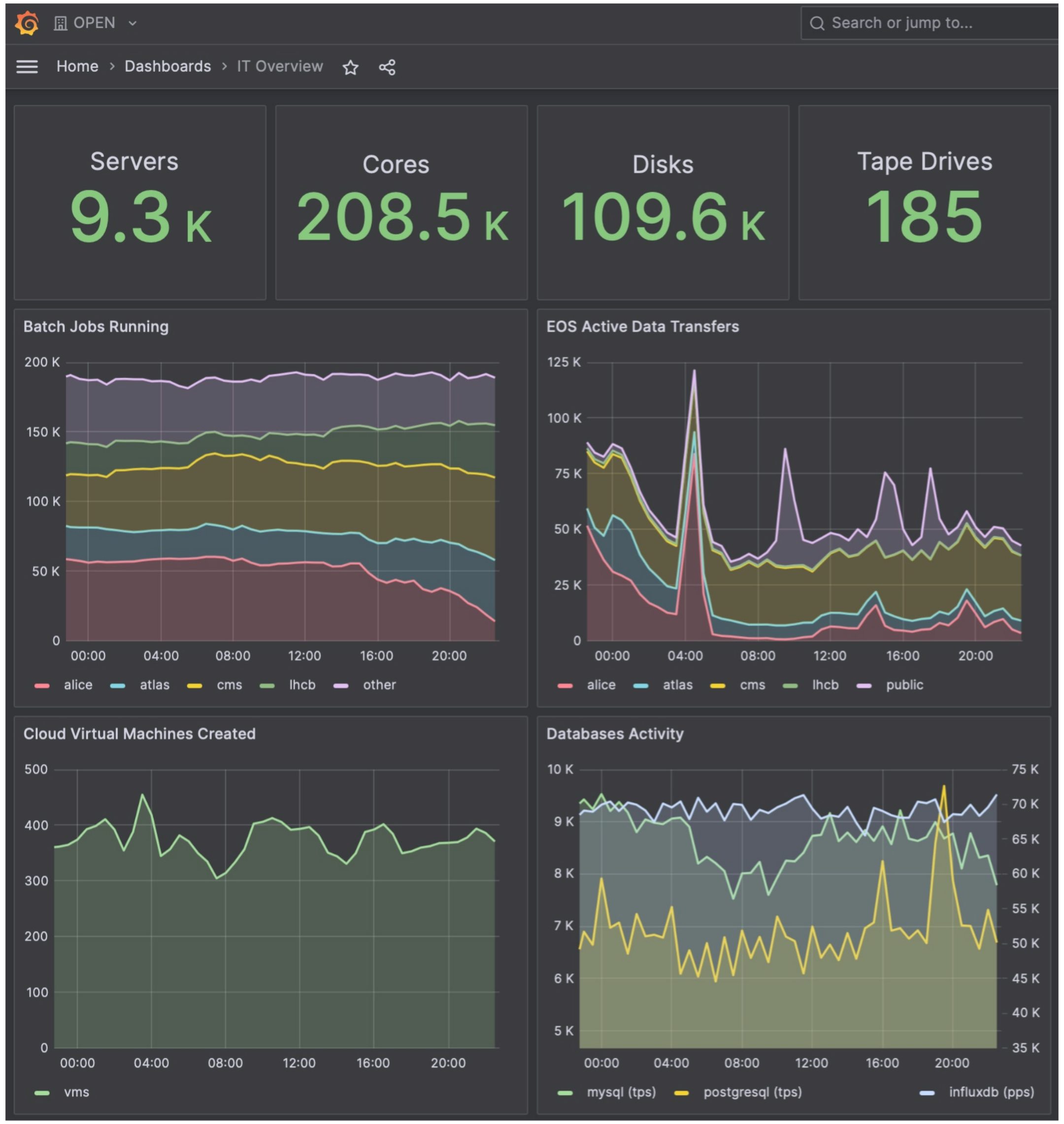
Data Access

Grafana (https://monit-grafana.cern.ch)

- Main data access and visualisation tool for MONIT
  - ~ 5k Grafana users / 2620 dashboards / 65 organizations
- **Alerting** functionality with **SNOW** integration
- Supports large number of different data source plugins

SWAN

- **HDFS** data access and analysis
- **Analytix** HDFS access and *Spark* integration


OPEN
Search or jump to...
Home › Dashboards › IT Overview
Servers 9.3 K
Cores 208.5 K
Disks 109.6 K
Tape Drives 185
Batch Jobs Running
alice atlas cms lhcb other
EOS Active Data Transfers
alice atlas cms lhcb public
Cloud Virtual Machines Created
vms
Databases Activity
mysql (tps) postgresql (tps) influxdb (pps)

Future Evolution & Further Ideas

Flume replacement

- OpenTelemetry Collector and Kafka Connect being evaluated for both the ingestion and Sink layers

Prometheus metrics long term storage (Grafana Mimir)

- Scalable system using S3 as storage backend
- MONIT deployment in K8S
- Pilot version already available

Processing platform for user jobs

- Providing functionality for the users to deploy/manage own jobs processing MONIT data

Summary

- **MONIT is a scalable Monitoring infrastructure**
  - Using *Kafka* as backbone of the service
  - Provides data processing capabilities
  - Different data storage per document type and user requirements
- **MONIT as client of the Hadoop Service**
  - Heavy user of the **Analytix** and **HadoopQA** clusters
    - **HDFS** for data storage
    - **YARN** for *Spark* job processing
- **Spark processing jobs are crucial part of the MONIT infrastructure**
- **MONIT evolution towards OpenTelemetry standards**
  - *Prometheus* long term storage
  - *OpenTelemetry* format and collector evaluation

Thank you!

SNOW: Monitoring Service
Mattermost: MONIT
Docs: https://monit-docs.web.cern.ch/

home.cern