



# MANAGING BUILD INFRASTRUCTURE AT ALICE USING HASHICORP NOMAD

COMPUTING IN HIGH ENERGY PHYSICS 2023, NORFOLK, VA

Timo Wilken<sup>1 2</sup>    Giulio Eulisse<sup>2</sup>

9 May 2023

---

<sup>1</sup>E-mail: [timo.wilken@cern.ch](mailto:timo.wilken@cern.ch)

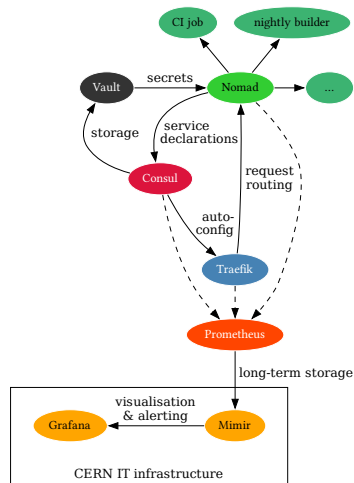
<sup>2</sup>for the ALICE Collaboration

# WHAT DO WE BUILD?

- ▶  $O^2$ : software suite for data-taking, physics analysis, Monte-Carlo simulation, ...
  - ▶ split across multiple repositories:  $O^2$ ,  $O^2$ Physics, [QualityControl](#) and more
- ▶ Run 2 software still maintained for analysing old data
- ▶ [alidist](#): pseudo-distribution of  $O^2$  dependencies
  - ▶ designed to function on top of recent versions of CentOS, Alma, Ubuntu, MacOS, ...
- ▶ nightly release builds, continuous integration (CI) compilation checks, unit and integration tests
- ▶ 1 non-trivial CI check completed every 2 minutes, on average
  - ▶ plus lots of fast rebuilds where nothing has changed
- ▶ ...all on 5+ platforms (mostly) through containerization

# ARCHITECTURE OVERVIEW

- ▶ resources: 600 CPU cores + 1.7 TiB memory
- ▶ Nomad, Consul, Vault from Hashicorp, designed to complement each other
- ▶ **Nomad**: allocates jobs to machines; resource accounting
  - ▶ long-running jobs: release builders, custom CI
  - ▶ web services: user account admin, tarball servers
  - ▶ scheduled jobs: repository maintenance and cleanup
- ▶ **Consul**: generic key/value store and DNS
  - ▶ job discovery: \*.service.consul DNS
  - ▶ **Traefik** auto-config for web access
  - ▶ job monitoring: simple health checks
- ▶ **Vault** stores secrets, using Consul as backend
- ▶ metrics of the whole cluster stored and visualised



# REASONS FOR SWITCHING AWAY FROM MESOS AND AURORA



The screenshot shows the Apache Aurora web interface. At the top, there's a breadcrumb trail: `mesos / mesosci / prod / rpm_creation_el8.x86_64`. Below this, there are buttons for 'Active tasks (1)', 'Completed tasks (0)', and 'All tasks'. The main section is titled 'Configuration Overview' and shows '0' instances. A table titled 'configuration details for instances 0' lists various parameters:

resources	cpu	1 core(s)
ram	4.00 GiB	
disk	100.00 GiB	
constraints	dedicated:/nocompile	
production	true	
tier	preferred	
service	true	
container	image	alisp/sic8-builder:latest

Below the table is a 'hide config' link and a table with columns 'Instance', 'Status', and 'Host'.

Instance	Status	Host
0	+ 25 days ago - RUNNING	alientest03.cern.ch

- ▶ previous stack: [Mesos](#) + [Marathon](#) + [Aurora](#)
- ▶ Aurora not intensively developed any more
- ▶ requires Python 2 (EOL since 2020) on server and developers' machines
  - ▶ difficult to install, deploy and maintain
- ▶ some features difficult to integrate with or nonexistent
  - ▶ autoscaling (or even manual scaling without restarts of all jobs)
  - ▶ difficult to keep build caches “hot”
  - ▶ little monitoring and alerting integration

## IMPROVEMENTS WITH NOMAD + CONSUL + VAULT

- ▶ simple deployment: static binary + systemd/launchd service + configuration = 3 files
- ▶ first-class support for web services: health checks, autoconfiguration
- ▶ more secure secrets management
- ▶ excellent monitoring & alerting support through Prometheus
  - ▶ resource use statistics (CPU, memory, disk)
  - ▶ alerts when build machines are unavailable or have problems
- ▶ ...more features, for deeper future integration

# WEB SERVICES: HEALTH CHECKS & TRAEFIK AUTOCONFIGURATION

The screenshot displays the Nomad web interface for a service named `process-pull-requests-http`. The configuration is shown in a code editor on the left, with two sections highlighted by red boxes and arrows pointing to the corresponding UI elements on the right.

```
57 service {
58   name = "${JOB}"
59   port = "http"
60   tags = [
61     # Strip a /github prefix off the URLs passed to this service.
62     "traefik.http.routers.github.rule=Host('ali*.cern.ch') && PathPrefix('/github')",
63     "traefik.http.routers.github.middlewares=github-stripprefix",
64     "traefik.http.middlewares.github-stripprefix.stripprefix.prefixes=/github",
65   ]
66
67   check {
68     type = "http"
69     port = "http"
70     path = "/health"
71     interval = "20s"
72     timeout = "5s"
73     initial_status = "warning"
74   }
75 }
```

**Left Panel (Service Details):**

- Service Name:** `process-pull-requests-http`
- Node:** `alibuild`
- Health Checks:** A table showing the health status of the service.
- Serf Health Status:** A table showing the health status of the node.
- Service: "process-pull-requests-http" check:** A table showing the health status of the service.

**Right Panel (Router Details):**

- Router Details:** Shows the configuration for the `github-stripprefix` router, including the rule `Host('ali*.cern.ch') && PathPrefix('/github')`.
- TLS:** Shows the TLS configuration for the router.
- Middlewares:** Shows the configuration for the `github-stripprefix` middleware, including the rule `Host('ali*.cern.ch') && PathPrefix('/github')`.

Red arrows indicate the mapping from the configuration code to the UI elements:

- From the `tags` block to the **Router Details** and **Middlewares** sections.
- From the `check` block to the **Service: "process-pull-requests-http" check** section.

## MONITORING EXAMPLE: NIGHTLY BUILD PERFORMANCE

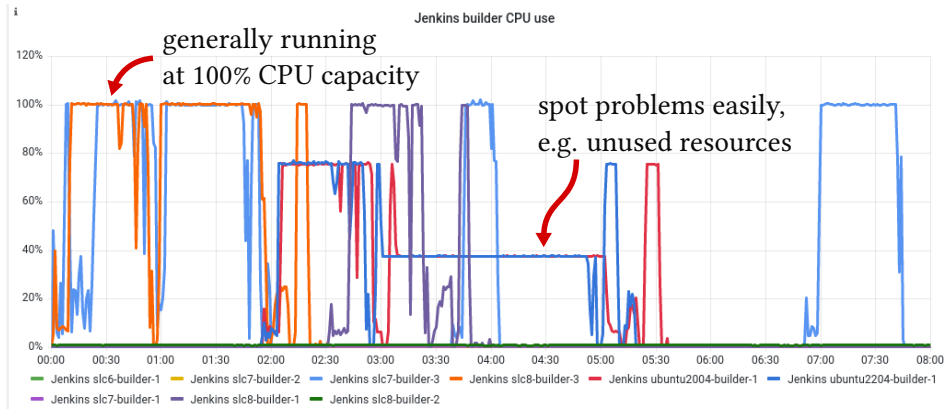


FIGURE: CPU use of a sequence of nightly builds as a fraction of total allocated CPU resources (usually the entire VM).

## MONITORING EXAMPLE: NIGHTLY BUILD PERFORMANCE

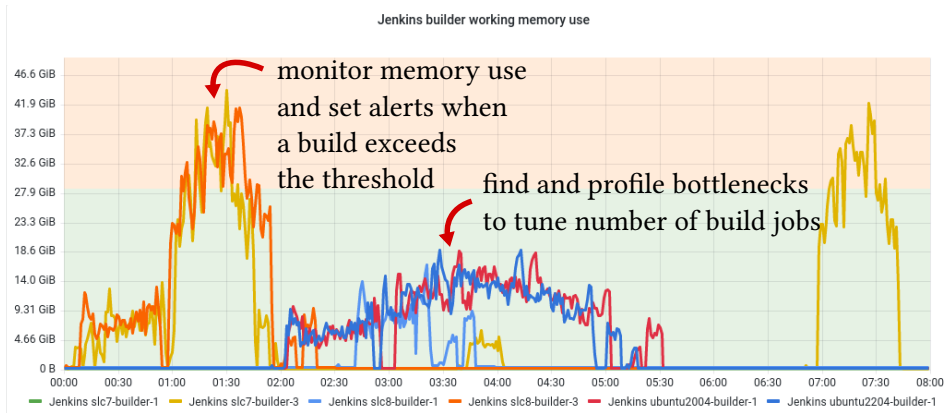


FIGURE: Working memory (RSS) use of a sequence of nightly builds. Total available memory on a typical build VM in green.



# ROUGH EDGES

1. Nomad's handling of disk space allocation
  - ▶ restarting daemon with non-empty disk confuses Nomad's accounting
  - ▶ can cause scheduling issues much further down the line
  - ▶ must manually clean up the node and restart the Nomad agent process
2. integration with CERN single sign-on
  - ▶ by default: token authentication with Nomad/Consul/Vault
  - ▶ could integrate SSO with Vault, which would then issue Nomad/Consul tokens

## FUTURE WORK INTEGRATING BUILD INFRA WITH NOMAD

- ▶ “true” autoscaling, based on real-time demand
  - ▶ manual scaling already much smoother than previously: build caches are kept most of the time, existing builders uninterrupted
  - ▶ remaining challenge is cache invalidation: scaling often invalidates multiple gigabytes of cached builds
- ▶ temporary configuration (e.g. for testing software deployment) through Consul instead of text files
- ▶ get build secrets from Vault only when needed, instead of storing them in env variables and relying on sanitisation during build