ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-THPDP064

SELECTING A LINUX OPERATING SYSTEM FOR CERN ACCELERATOR CONTROLS

A. Radeva, F. Locci, J.M. Elyn, T. Oulevey, M. Vanden Eynden
CERN, Geneva, Switzerland

Abstract

Changing the operating system (OS) for large heterogeneous infrastructures in the research domain is complex. It requires great effort to prepare, migrate and validate the common generic components, followed by the specific corner cases. The trigger to change OS mainly comes from Industry and is based on multiple factors, such as OS end-of-life and the associated lack of security updates, as well as hardware end-of-life and incompatibilities between new hardware and old OS. At the time of writing, the CERN Accelerator Controls computing infrastructure consists of 4000 heterogeneous systems (servers, consoles and front-ends) running CentOS 7. The effort to move to CentOS 7 was launched in 2014 and deployed operationally 2 years later. In 2022, a project was launched to select and prepare the next Linux OS for Controls servers and consoles. This paper describes the strategy behind the OS choice, and the challenges to be overcome in order to switch to it within the next 2 years, whilst respecting the operational accelerator schedule and factoring in the global hardware procurement delays. Details will be provided on the technical solutions implemented by the System Administration team to facilitate this process. In parallel, whilst embarking on moving away from running Controls services on dedicated bare metal platforms towards containerization and orchestration, an open question is whether the OS of choice, RHEL9, is the most suitable for the near future and if not what are the alternatives?

CONTROLS COMPUTING INFRASTRUCTURE

From a computing perspective, the CERN Control System is structured across three physical layers (Fig. 1):

1. The top (or client) tier consists of computers installed in the CERN Control Center (CCC), used by operations teams and equipment experts to run high-level graphical applications for accelerator control.
2. The middle (or business) tier, is comprised of powerful, high-availability servers, running the core control systems services which the high-level applications interact with.
3. The lower (or Front-End) tier is made of embedded computers (FECs) that execute real-time applications, interfacing with electronic boards to control and monitor accelerator devices.

Configuration aspects are handled by a central Controls Configuration Service (CCS) which is built around a relational database (CCDB) [1].

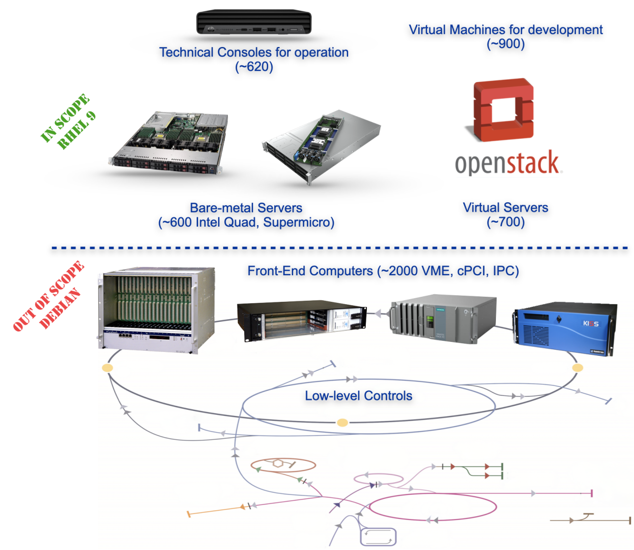

Figure 1: Computing layers in CERN's Control System.

OPERATING SYSTEM LIFE CYCLE

To date, all computers within CERN's accelerator control system are based on a single Linux Operating System (OS). Currently, this is the CERN Community Enterprise Operating System (CC), which is based on CentOS Linux 7, which in-turn, is a downstream derivative of Red Hat Enterprise Linux (RHEL) 7. CERN's IT department closely follows the Red Hat OS life cycle (Fig. 2), prepares the corresponding CERN-specific distribution, and provides upstream support.

Figure 2: Red Hat Enterprise Linux Life Cycle.

The IT department communicates with key industry and Open Source Software Community actors for guaranteeing that the necessary specific packages are rebuilt and made available for the experts in CERN's Accelerators and Technology Sector.

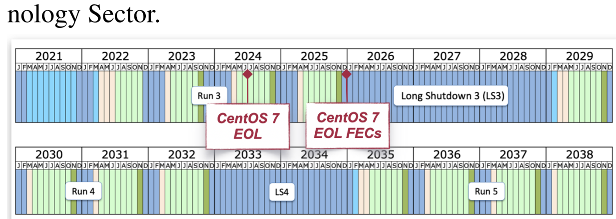

Figure 3: CERN's Long-term Accelerator Schedule.

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-THPDP064

The guaranteed OS stability and support for a period of up to 10 years is an important factor, as it allows sufficient time to prepare operating system upgrades that align with CERN's long-term accelerator operation schedule (Fig. 3).

The last major OS change in the control system was the migration to CentOS 7, which was triggered 9 years ago. The OS validation phase took 2 years, leading into a progressive migration of all technical consoles, servers and FECs. This was followed by 7 years of stable production (Fig. 4).

Figure 4: Linux OS Life Cycle for Accelerator Controls.

Factors Driving the OS Selection Process

The factors triggering a change of OS and driving the choice of a replacement are mainly external and coming from the Industry:

- Microprocessor chipset deprecation
- OS end-of-support and end-of-life
- Security
- Support by the Open Source Software Community
- Support bricefor third-party industrial controls software solutions

Microprocessor chipset deprecation, in particular for the high-availability server platform and the CCC consoles is a key factor for triggering an OS change. The CERN Controls System Administration team anticipated the impact of this situation and started the evaluation of the next natural candidate OS, i.e. CentOS Linux 8, at the end of 2020.

In 2021, a change in the LHC schedule caused a misalignment between the start of LHC Long Shutdown 3 (starting 1 year later at the end of 2025) and the CentOS Linux 7 end-of-life (EOL), i.e. June 30, 2024, which falls in the middle of the physics run (Fig. 3).

Moreover, at the end of 2020, the CentOS Project, in coordination with Red Hat, announced [2] that it would shift full investment to CentOS Stream, to become the upstream development platform for upcoming Red Hat Enterprise Linux releases. This played a decisive role in the final choice of the next OS. With a shortened life cycle (5 years), CentOS Stream 8 was no longer considered as a natural CentOS Linux 7 successor.

At the same time, Red Hat published a clear life cycle planning for all future distributions with precise release and EOL dates. The RHEL versions 8 and 9 planning include up to 13 years of full maintenance and extended support. This aligns far better with CERN's accelerators schedule and the operational constraints in comparison to the reduced, 5 year lifetimes of CentOS Stream 8 and 9.

Taking into consideration CERN's long-term accelerator schedule and the Red Hat support strategy change, the IT department revisited the IT strategy for the future Linux OS and provided RHEL 8 and 9 Linux distributions for the CERN user community (Fig. 5).

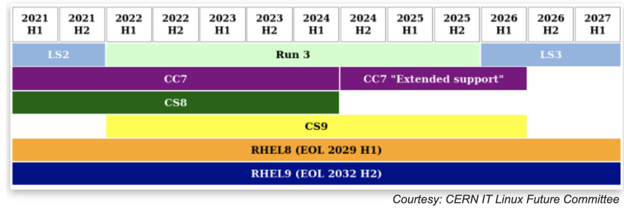

Figure 5: CERN IT Future Linux Strategy.

Choosing the Next OS for Controls

In May 2022, directly after the RHEL 9 release, CERN's Controls System Administration team performed 3-months of intensive evaluation of RHEL 9. Open source packages not available in the repositories yet, were recompiled for an early internal validation. The Open Source Software community followed up quickly and within a few months provided the necessary packages. However, some remaining unsupported packages still had to be recompiled and packaged internally.

Following a positive evaluation phase, the final choice of RHEL 9 as next OS was made in September 2022. This was only possible within this time frame, thanks to the prior 8-months of CentOS Stream 8 and 9 validation, from which RHEL 9 is derived. CentOS Stream 9 allowed system administrators to validate key components before the RHEL 9 final release. The availability and compatibility of essential libraries could be assessed, and commercial applications could be validated in advance. This proactive approach allowed for early engagement with support and sales teams to seek assistance in the validation of upcoming versions. An early concern, for example, was the absence of TLS 1.0 support, which could be mitigated or worked-around before the official RHEL 9.0 launch. The full timeline for the OS evaluation and selection is shown in Fig. 6.

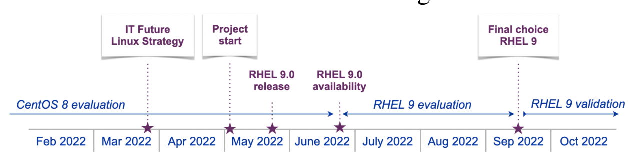

Figure 6: OS evaluation and selection timeline.

Project Scope

In 2022, Red Hat made another important microprocessor deprecation announcement, affecting about 65% of the operational Front-End tier embedded system microprocessors. Red Hat's decision to increase the supported x86-64 micro architecture level to x86_64v2 in RHEL 9, and x86_64v3 in RHEL 10, means that old, but perfectly functioning controls hardware and associated devices (i.e: PCI cards), need to be replaced in order to continue using RHEL.

This abrupt Red Hat announcement and the huge replacement cost it implies, led the CERN Controls teams to revisit the OS strategy. It was decided to move forwards as planned

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-THPDP064

with RHEL 9 for the high-availability servers and CCC consoles, and to take a separate OS strategy for the embedded systems, selecting the Debian OS [3, 4]. As a consequence, the CERN accelerator control system is now being moved away from the single OS approach that has prevailed since the beginning of the LHC-era.

OS INTEGRATION & VALIDATION

The integration and validation of a new OS in the CERN accelerator controls ecosystem is a complex process which requires a good level of automation. Moreover, since the last major OS upgrade was made in 2014, it was the right time to embrace state-of-the art practices and replace obsolete home-made system administration tools by well-established standard solutions.

Automation and Tools

Configuration Management Based on valuable experience gained over many years, the System Administration team initiated a major refactoring and simplification of the Linux configuration infrastructure (Fig. 7). The main objectives were to make it distribution-agnostic and enable support for multiple distributions, in parallel. The installation and configuration of the OS and supporting packages is managed with Ansible, which offers flexibility, a shallow learning-curve and easy management of large infrastructures [5]. Instead of managing a single Ansible instance for all Linux distributions, the System Administration team opted for a dedicated instance for RHEL 9, designed from scratch, with potential future distribution changes in mind. In this new model, tools were developed to automate and pipeline the OS migration process. Another important area of improvement concerned hardware and software configuration, for which a single source of information, i.e. the CERN IT central network database (LanDB), has been integrated. While LanDB is used for CERN-wide networked device configuration, Controls-specific configuration is managed using the aforementioned CCS. As such, an automatic synchronisation mechanism was implemented between LanDB and the CCDB. In-turn, the CCS is used to facilitate the planning and follow-up of the actual migration of the 2,000 computers to RHEL 9, and it is also foreseen to support future inventory management.

Plain Containerisation Many factors have contributed to the widespread adoption of containerization and it has become a de-facto standard for companies to deploy software. CERN is no exception and took advantage of the OS transition period to invest definitively in the use of container technology. Containerization can be used in two manners:

1. Plain Containerization : deploying containerized software directly on the existing system stack.
2. Container Orchestration : dedicating generic resources and a cluster-based approach to manage the full container life cycle.

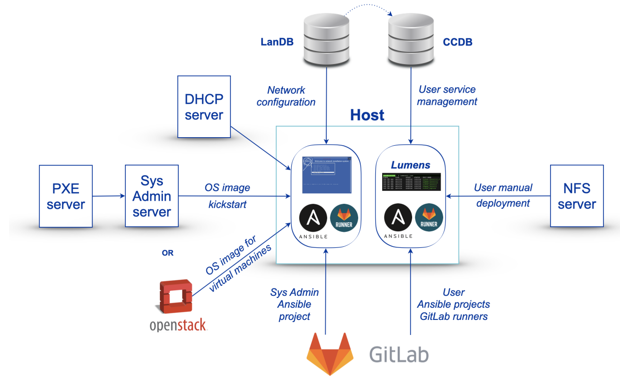

Figure 7: Linux ecosystem for Controls.

Although the second solution offers far more advantages, in terms of reliability, scalability and resilience, the CERN Controls group opted to already start with plain containerization on CentOS 7. This allowed development teams to quickly begin a switch from delivering monolithic applications to container-ready, micro-services. In-turn, the System Administration team focused on providing support for containers in the new OS environment, then preparing an orchestration solution [6]. Podman [7] was chosen as a lightweight (daemonless architecture) alternative to Docker, to run containers without root privileges (rootless), improving the security in the production environment [8].

Containerization can simplify OS migration, as the containerized software can, by definition, run on any OS. As such, containerization has the potential to artificially extend the viability of unsupported software. While this is an option, there are many drawbacks, including security aspects. As such, it is advised that developers ensure that they stay actively engaged in managing the evolution of their software, even if it runs in a container.

Process Management The Control System is made of many software processes that need to be deployed, managed and monitored. The LUMENS (Linux Units ManagemENt System) tool was developed to respond to these needs. In addition, the tool needs to:

- Be well-integrated with the aforementioned CCS.
- Be language and technology agnostic.
- Cover the needs of all categories of computers (consoles, servers, and FECs).

Internally, LUMENS relies on systemd [9], the de-facto standard system and service manager for Linux. However, LUMENS provides its own CLI and API, tailored to the CERN environment. It also provides process monitoring and alerts, thanks to integration with COSMOS, the Controls Infrastructure Monitoring System [10].

As depicted in Fig. 8, to use LUMENS, the software service to be managed needs to be declared in the CCS, and then a systemd unit file can be generated. The LUMENS CLI (Command Line Interface) provides commands to start and stop the software services remotely, with host access rights being managed by CERN egroups [11]. Because the

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-THPDP064

services are declared centrally in the CCS, COSMOS is able to automatically perform basic monitoring to check whether a process is up or down.

In summary, LUMENS is a lightweight software, written in Python, and it allows advanced users to benefit from the full power of systemd when required, including management of plain container services.

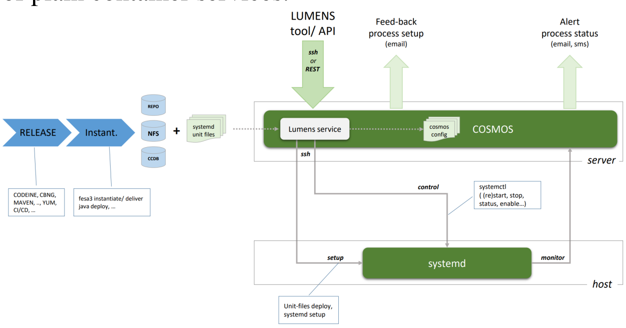

Figure 8: LUMENS software architecture.

OS MIGRATION STRATEGY

The OS migration for large, heterogeneous computing infrastructure, running 24/7, such as for CERN's control system, is a complex and time-consuming process. The actual change from CentOS Linux 7 to RHEL 9 can only be done with an installation from scratch, which increases the required down-time and has an associated risk. To counter this requires a duplication of computing resources for validation purposes and for keeping a backup solution readily available.

The biggest challenge faced by the System Administration team is the short accelerator Year-End Technical Stop (YETS) period, which lasts less then six weeks, but during which many critical systems have to be migrated and validated. This is further complicated by the fact that in addition, physical hardware upgrades can be necessary in some cases.

Computer Categorisation

To facilitate the scheduling of the OS migration, all computers are grouped into one of several categories, based on their type and usage:

- Technical consoles: used for operations or by equipment experts
- Bare-metal servers: back-end servers and dedicated PCs
- Virtual servers
- Virtual machines: for application development
- Virtual machines: for Front-End software development
- Front-End Computers: out of scope

For each category the functional requirements and operational constraints were specified and discussed with the stakeholders.

Technical Consoles and Validation

The Technical consoles category was validated first. The subsequent OS migration was launched and performed progressively during the accelerator running period. Key experts from the Operations team participated in the early validation of RHEL 9 in the operational environment. The major challenges for the System Administration team were to follow up the validation of all software application frameworks and specific software applications used in Operation with the developers responsible. This collaboration extended to both CERN developers and industry partners. It also encompassed the resolution of numerous complex legacy configurations and applications.

A particularly interesting case is the use of a container-based technical solution provided for the numerous WinCC OA client applications which needed to run at the same time in multiple heterogeneous client environments. For example, WinCC OA client applications needed to run on CentOS Linux 7 and RHEL 9 consoles, but there was an incompatibility issue between the WinCC OA version 3.16 and the RHEL 9 OS. The transitional container-based solution is depicted in Fig. 9.

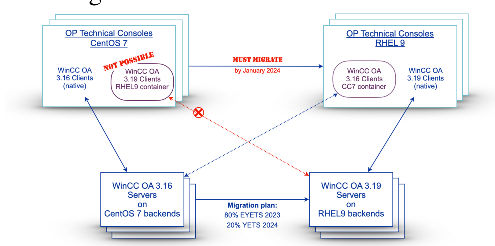

Figure 9: Transitional solution for WinCC OA applications.

Server Inventory and Migration Plan

Servers are at the heart of the infrastructure and are the most critical category. A detailed inventory was performed for the 600 machines, covering the following aspects:

- Hardware upgrade necessity
- Network configuration
- Software configuration and deployment
- Services
- System availability needs and target migration period
- Migration, validation and rollback strategy
- Responsibilities and roles

After analysing the Servers inventory, a detailed migration plan was elaborated, grouping the servers into batches based on the accelerator shutdown and startup dates. Important milestones, dependencies and constraints were also integrated (Fig. 10).

Migration Process

For every server or group of servers, before starting the validation, users of the machine and System Administrators must work in close collaboration to follow an iterative migration process shown in Fig. 11.

To facilitate and optimise the migration process the following key principles have been defined:

- Provide a stable testbed covering all known categories.
- Provide Openstack cloud images.

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-THPDP064

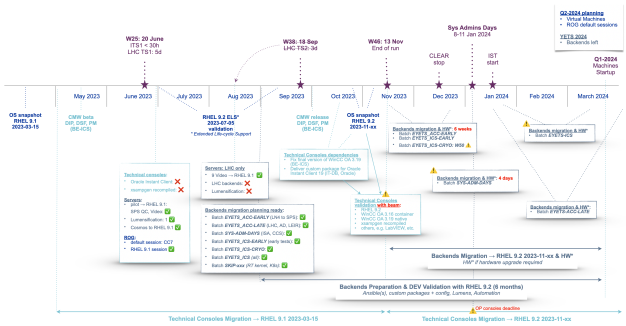

Figure 10: OS Migration timeline.

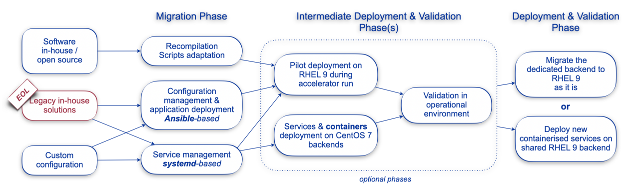

Figure 11: Iterative migration process.

- Rely on the CI/CD solution based on Ansible and GitLab.
- Perform early pilot deployments in operation during the run.
- Perform early migration from legacy service management to LUMENS.
- Involve key users in the process, e.g. Operations.
- Address large critical systems first, e.g. WinCC OA systems for cryogenics, vacuum control etc.
- Perform massive migration of platform-independent Java servers at an early stage.
- Tackle the specific and time-consuming cases later.
- Provide tools and automatic generation of configuration and deployment for Ansible and LUMENS services;
- Pipeline the migration of multiple systems in parallel.

CONCLUSION

Selecting the best OS and organizing the migration of thousands of operational CERN Accelerator computers is a major endeavor. It requires a very close follow-up of the fast evolving IT landscape and the preparation of a sustainable strategy, compatible with the demanding LHC schedule for the years to come. The decision to use RHEL 9 for the CERN CCC consoles and for the high-availability servers will ensure guaranteed support until 2032. Nonetheless, the last microprocessor deprecation roadmap announced by Red Hat had a major impact on CERN's OS strategy for embedded systems for which the target microprocessor lifetime is in the order of 15 years. An alternative strategy for those systems has been put in place, based on the Debian OS, giving CERN more flexibility in terms of evolution and upgrades.

ACKNOWLEDGEMENTS

CERN's Controls System Administration team would like to thank the partners from the IT department for the close collaboration in the process of the selecting the future Linux OS for CERN Accelerator Controls. This solid partnership is a key factor in taking the right decisions and going forward for the implementation with confidence. Thanks also go to the software service providers and the operators, who helped validate the pilot technical solutions and overcome the incompatibilities between RHEL 9 and the legacy software still in operation running on CentOS 7.

REFERENCES

[1] L. Burdzanowski *et al.*, "CERN Controls Configuration Service — A challenge in usability", in *Proc. ICALEPCS'17*, Barcelona, Spain, Oct. 2017. doi:10.18429/JACoW-ICALEPCS2017-TUBPL01

[2] CentOS Project shifts focus to CentOS Stream, https://blog.centos.org/2020/12/future-is-centos-stream/

[3] F. Vaga, "Front-End Computer (FEC) Operating System Renovation", in *ICALEPCS 21 : Linux Distrib. Ecosyst. Evol. Control Syst. Workshop*, 2021. https://indico.cern.ch/event/1084050/contributions/4566329/attachments/2328106/3966540/20211014-icalepcs.pdf

[4] F. Vaga, "ATS - Linux for FECs and SoCs", in *3rd CERN System-on-Chip Workshop*, CERN, Geneva, Switzerland, 2021. https://indico.cern.ch/event/1253805/#36-ats-linux-for-fecs-and-socs

[5] Ansible, https://www.ansible.com

[6] B. Copy *et al.*, "An Update on the CERN Journey from Bare Metal to Orchestrated Containerization for Controls", presented at ICALEPCS'23, Cape Town, SA, 2023, paper TH2AO03, this conference.

[7] Podman, https://podman.io

[8] R. Voirin, T. Oulevey, and M. Vanden Eynden, "The State of Containerization in CERN Accelerator Controls", in *Proc. ICALEPCS'21*, Shanghai, China, Oct. 2021, pp. 829–834. doi:10.18429/JACoW-ICALEPCS2021-THBL03

[9] systemd, https://systemd.io

[10] F. Locci, J. P. Palluel, F. Ehm, L. Gallerani, J. Lauener, and R. Voirin, "CERN Controls Open Source Monitoring System", in *Proc. ICALEPCS'19*, New York, NY, USA, Oct. 2019, pp. 403. doi:10.18429/JACoW-ICALEPCS2019-MOPHA085

[11] A. Corman *et al.*, "CERN's Identity and Access Management: A journey to Open Source", in *CHEP 2019*, Adelaide, Australia, 2019, p. 03012. doi:10.1051/epjconf/202024503012