Large scale dynamic web deployment for CERN, experience and outlook

Rajula Vineet Reddy[1,*], *Joachim* Valdemar Yde[1,**], *Francisco* Borges Aurindo Barros[1], *Carina* Antunes[1], and *Andreas* Wagner[1]

[1]CERN

Abstract. CERN hosts more than 1200 websites essential for the mission of the Organization, both for internal and external collaboration and communication, as well as public outreach. The complexity and scale of CERN's online presence is very diverse with some websites, like home.cern, accommodating more than one million unique visitors in a day. However, regardless of their diversity, all websites are created using the Drupal content management system (CMS), and are self-hosted directly in the CERN Data Center on a dedicated infrastructure that runs on Kubernetes. Workflows like provisioning, deleting, cloning, upgrading, and similar are fully automated and managed by a customized Kubernetes controller. By leveraging the custom controller, the infrastructure has proven highly reliant with minimal, manual intervention necessary. In order to further automate deployments and improve governance, a customized version of Drupal called the CERN Drupal Distribution is implemented. Supported by end-to-end integration tests and automated browser simulation, this setup enables the propagation of security and feature updates seamlessly to all websites without any downtime.

This paper outlines the architecture which allows building, testing, and distributing updates to a large number of websites without any downtime. Furthermore, it presents experiences and learnings from managing such a service at CERN with limited resources.

1 Introduction

Collaboration lies at the heart of CERN's mission and one critical aspect of this collaboration is facilitated through Content Management Systems[1]. CERN hosts thousands of websites catering to different purposes for the global physics community. All of these websites are hosted by CERN on premises and offered as a service by the IT department. This paper presents a continued perspective following an earlier publication at CHEP 2021 [2] which details the initial design and implementation plan for the infrastructure supporting CERN's content management systems.

Since the establishment of the aforementioned infrastructure and automation built around it, the system is serving the community in production ever since. The design and the infrastructure have undergone multiple iterations of improvements during this time. The infrastructure has proven quite efficient, scalable, secure and improved user experience.

---

*e-mail: rajula.vineet.reddy@cern.ch
**e-mail: joachim.valdemar.yde@cern.ch

Figure 1. *Production websites organized by departments.* Note: "Other" here refers to websites owned by users who do not represent a department according like external collaborators or alumni or users

The reflections presented in this paper aspire to share a wealth of knowledge and insights, that can be applied to any organisation and institution facing similar challenges in managing complex content management systems.

2 Blueprints for a successful large scale dynamic web infrastructure

2.1 Requirements

2.1.1 CERN's Online Presence

CERN has a strong online presence and its websites are distributed widely. As of Tuesday 9th May, 2023, the Drupal infrastructure alone, hosts a total of 837 production websites.

As an organisation, CERN is structured into increasingly smaller units. At the higher level Departments exist, which are split into groups and further divided into smaller sections. CERN's websites are owned and maintained by members from various departments of CERN. Figure 1 provides a breakdown of the 837 production sites by department *i.e.* total of websites owned by users associated with each department. It can be seen that among the departments one stands out: the majority of the websites belong to the EP (Engineering Physics) department, which includes members of large experiments like CMS, ATLAS, ALICE and LHCb.

2.1.2 Audience

The websites serve an essential role in supporting daily operations, publishing press releases, making announcements, facilitating collaboration, and more. Consequently, they cater to a diverse audience both internal and external to the organisation. Internal audience includes staff members, users and collaborators, whereas the external audience encompasses visitors, members of the press, member states, among others.

In essence, these websites act as a dynamic and indispensable interface between CERN and the community it serves, ensuring that information flows efficiently and effectively to all stakeholders.

2.1.3 Centralized Infrastructure

The centralized management of all websites within the infrastructure not only guarantees the consistent use of up-to-date software and the application of security patches but also streamlines the process of ensuring that each website complies with design and communication guidelines. This approach enhances overall website performance, minimizes vulnerabilities, and fosters a cohesive and professional online presence for the organization. It enables efficient monitoring and maintenance, promoting a seamless user experience while upholding essential standards and protocols.

2.1.4 Unique requirements

The diverse array of websites hosted on the infrastructure presents a considerable challenge. Each site comes with its unique set of demands, spanning resource allocation, design intricacies, feature sets, target audiences, and traffic patterns. Moreover, the varying levels of technical proficiency among site administrators and developers introduce an added layer of intricacy. Consequently, managing and maintaining the infrastructure necessitates a highly adaptable and nuanced approach, capable of accommodating these multifaceted requirements while ensuring optimal performance and security across the board.

2.1.5 Standardized configuration and maintenance

To accommodate the distinct requirements and specific needs of each website, the infrastructure enforces standardization in terms of site configuration and maintenance processes. Configuration, in this context, encompasses integration with various services such as databases, authentication systems, file storage, and web analytics. Furthermore, from an infrastructure perspective, the lifecycle and maintenance of websites, including provisioning, backups, restoration, and deletion, follow identical procedures across all sites. This standardized approach ensures consistency and efficiency while catering to the unique demands of each web presence.

2.2 Infrastructure

The majority of the infrastructure relies on free and open-source software. At its core, Drupal is used as the Content Management System, an open-source site building tool crafted in PHP. This core software is customized and enhanced by incorporating essential modules, themes, and automation building an element referred to as the *CERN Drupal Distribution*, which consists of three key sub-components:

Modules This component encompasses both upstream Drupal modules that provide common functionality across sites and custom CERN modules tailored for integrating internal services.

Themes The themes section includes a custom CERN theme that ensures a consistent look and feel across all sites.

Automation The distribution also incorporates server configurations, custom startup and maintenance scripts, which are subsequently integrated with Kubernetes.
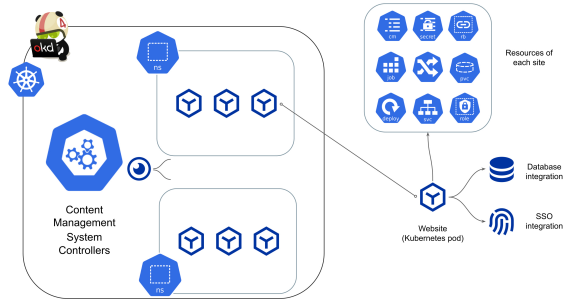
Figure 2. *Brief Infrastructure Overview showing the basic components*

Using a packaged approach such as the CERN Drupal Distribution, brings multiple benefits, namely:

Standardisation It simplifies the enforcement of standardized practices across all sites.

Centralized development Facilitates centralized development efforts, making it feasible to implement features or address issues across all sites simultaneously.

Design Guidelines Ensures that CERN's design guidelines[3] are both mandated and consistently managed across all sites through the themes included in the distribution

Modularity Allows for the addition of extra modules on top of existing ones through the *source to image* [4] image approach.

The distribution is then packaged into a container image, which is deployed on Kubernetes[5] through OpenShift OKD[6] platform.

The infrastructure primarily relies on controllers, with the Content Management System (CMS) controller being the central element. Illustrated in Figure 2 of the overview, the CMS controller actively monitors changes within all user namespaces via Custom Resources[7]. Each Namespace[8] corresponds to a primary website and is exclusively owned by a user and/or an admin e-group. Secondary websites, such as clones or test versions of the primary website, coexist within the same Namespace. Every website within a Namespace is associated with a Kubernetes Custom Resource Definition (CRD), which, in turn, governs multiple child resources[8], including Deployments, Configmaps, Secrets, Roles, RoleBindings, Jobs, Routes, PersistentVolumes, PersistentVolumeClaims, and Services.

Among these resources, the Kubernetes pod(s) serve as the actual operational component, housing various servers like Nginx[9], PHP-FPM[10], and WebDAV[11] as containers within the pod. These servers operate seamlessly with the assistance of our custom distribution.

Furthermore, the website CRD seamlessly integrates with other services, such as Databases and Authentication, through their respective controllers and CRDs.

Further details can be seen in the prior publication [2].

2.3 Releases

Given the infrastructure overview and the CERN custom distribution, rolling out a new release or running migration campaigns cluster wide for all websites is simplified. The process of rolling out a new release is depicted in figure 3 and figure 4.
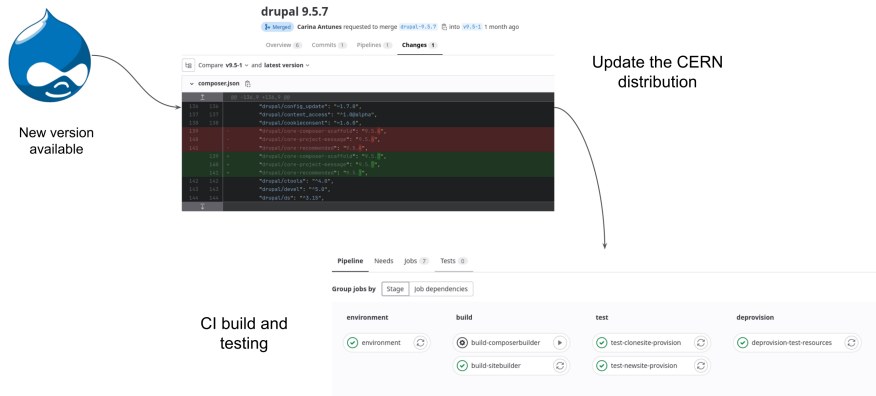
Figure 3. First part of the release process, showcasing steps involved in updating and building the CERN Drupal Distribution

1. When an upstream release becomes available, a merge request is initiated against the primary branch of the CERN custom distribution [12] . This request focuses solely on modifying the core versions specified in the *composer.json* file

2. After the merge request is created, a Continuous Integration (CI) pipeline is triggered to execute the changes. This pipeline includes tasks such as building and running tests, which involve operations like creating and replicating websites in a staging environment

3. Following these initial steps, an automated browser testing procedure is launched. This involves the creation of a new website and the assessment of its fundamental functionality using a web browser

4. Once the tests are successfully completed, and the changes are reviewed and approved, the merge request is merged. This action triggers the generation of a new image with a production tag. This image becomes accessible within the cluster through the *SupportedDrupalDistribution* CRD

5. The final step involves updating all websites to the new version. This is achieved by modifying the *version* field in the websites CRD for all websites. The controller detects this change and initiates the rollout of new deployments, ensuring a seamless transition without any noticeable downtime

2.4 User Interface

Site administrators have the authority to modify CRDs and execute tasks such as backup and restoration. To accommodate administrators with varying technical proficiencies, a user-friendly Web Interface is offered, known as the Web Services Portal. A view from the man-
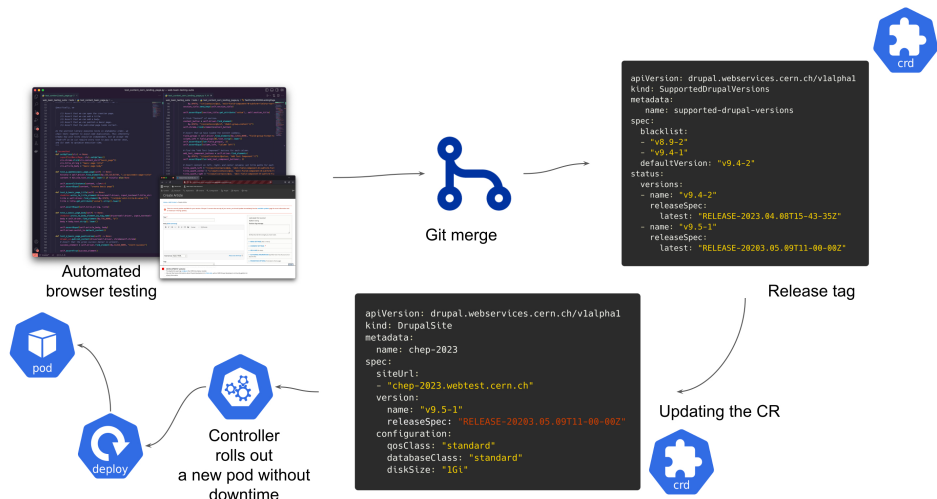
apiVersion: drupal.webservices.cern.ch/v1alpha1
kind: SupportedDrupalVersions
metadata:
    name: supported-drupal-versions
spec:
  blacklist:
  - "v8.9-2"
  - "v9.4-1"
  defaultVersion: "v9.4-2"
status:
  versions:
  - name: "v9.4-2"
    releaseSpec:
      latest: "RELEASE-2023.04.00T15-43-35Z"
  - name: "v9.5-1"
    releaseSpec:
      latest: "RELEASE-20203.05.09T11-00-00Z"

Automated
browser testing

Git merge

Release tag

apiVersion: drupal.webservices.cern.ch/v1alpha1
kind: DrupalSite
metadata:
  name: chep-2023
spec:
  siteUrl:
  - "chep-2023.webtest.cern.ch"
  version:
    name: "v9.5-1"
    releaseSpec: "RELEASE-20203.05.09T11-00-00Z"
  configuration:
    qosClass: "standard"
    databaseClass: "standard"
    diskSize: "1Gi"

pod

deploy

Controller
rolls out
a new pod without
downtime

Updating the CR

Figure 4. Second part of the release process, showcasing steps involved in releasing changes to the CERN Drupal Distribution to the websites

agement page for Drupal websites can be seen in Figure 5. Diverse options can be seen. Through the portal, users can perform a range of actions, including but not limited to:

- Accessing and reviewing their site's configuration and information
- Modifying site characteristics such as category, description, or transferring ownership
- Performing backups and restoring their site
- Modifying site URLs or adding aliases
- Creating test or clone sites by duplicating data from an existing site within the project

2.5 Traffic

The infrastructure, as briefly described earlier, has been operational for approximately two years. For the majority of websites, the typical traffic remains within expected levels, aligning with basic resource allocation, without any noteworthy anomalies. However, when it comes to the official CERN website, home.cern, it experiences unusual traffic surges during special LHC and CERN events. One such event occurred in July 2022 with the *start of LHC Run-3* [13]. As depicted in the figure 6, during this event, the website recorded an impressive influx of nearly one million unique visitors within just a couple of days.

3 Closing reflections

The experience lived over the extended period in which the described infrastructure has been in place in production, has proven how significantly it streamlines operations, effectively reducing the need for extensive manual interventions. This, in turn, simplifies the management and operation of the entire infrastructure, which is of key significance when working with a lean team and a growing number of users to support. Furthermore, upgrades are seamlessly
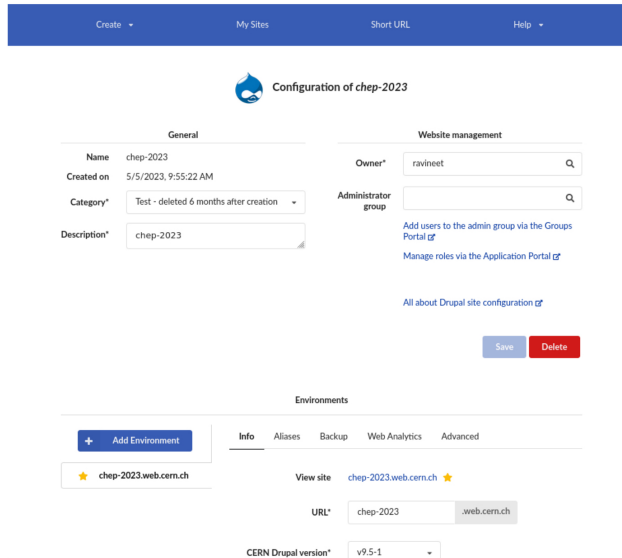
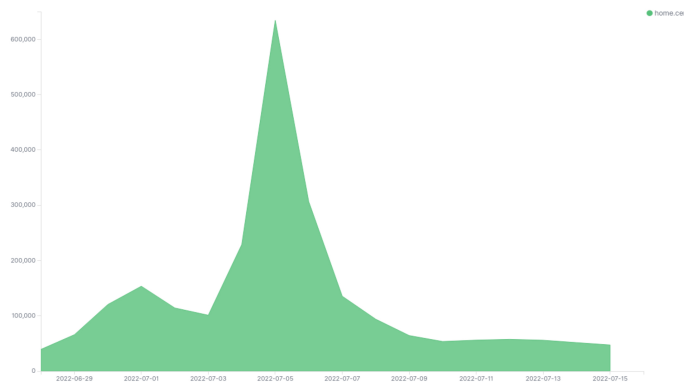**Figure 5.** Webpage for Drupal website management page in the Web Services Portal interface

Figure 6. Unique IP address received on home.cern during the week of LHC Run 3 in July, 2022. Peak at one million unique visitors.

automated, offering an enhanced level of transparency through continuous monitoring and timely alerts.

In addition, the introduction of this infrastructure has reduced vendor lock-in, by allowing to adapt in an easier manner to new Content Management Systems. This is possible with some additional development and refactoring of the Drupal components, however a large core fraction of the system can be reused.

Lastly, a transition as such presents an excellent chance to engage with the broader community, gathering valuable feedback that benefits everyone involved.

However, challenges should be noted as well. Namely the transition to this new infrastructure initially posed a learning curve for users, which was initially underestimated. Specially non technical users encountered additional challenges in adapting to new technologies and methodologies of working. Incorporating end-user perspectives into the implementation sooner should mitigate this problem, as well as focusing special attention to user facing documentation and training.

4 Source Code

At the time of publication, only the custom CERN Drupal Distribution is Open Source and available at https://gitlab.cern.ch/drupal/paas/cern-drupal-distribution.

Acknowledgements

References

[1] *Our mission*, https://www.home.cern/about/who-we-are/our-mission (2023), accessed: 21-09-2023

[2] Samaras-Tsakiris, Konstantinos, Vineet Reddy, Rajula, Borges Aurindo Barros, Francisco, Alvarez Fernandez, Eduardo, Wagner, Andreas, EPJ Web Conf. **251**, 02064 (2021)

[3] *CERN Design Guidelines*, https://design-guidelines.web.cern.ch/ (2023), accessed: 05-12-2023

[4] *Openshift source to image · github*, https://github.com/openshift/source-to-image (2023), accessed: 19-09-2023

[5] *Kubernetes*, https://kubernetes.io (2023), accessed: 21-09-2023

[6] *OKD*, https://www.okd.io (2023), accessed: 21-09-2023

[7] A. Raul, *Cloud Native with Kubernetes* (Packt Publishing, 2021)

[8] M. Hausenblas et al., *Programming Kubernetes* (O'Reilly Media, Inc., 2019)

[9] *Nginx*, https://www.nginx.com (2023), accessed: 21-09-2023

[10] *FastCGI Process Manager (FPM)*, https://www.php.net/manual/en/install.fpm.php (2023), accessed: 21-09-2023

[11] L. Dusseault, *WebDAV: Next-Generation Collaborative Web Authoring: Next-Generation Collaborative Web* (Prentice Hall, 2003)

[12] *Drupal / paas / cern-drupal-distribution · gitlab* (2023), https://gitlab.cern.ch/drupal/paas/cern-drupal-distribution

[13] *LHC Run 3: physics at record energy starts tomorrow*, https://home.cern/news/news/physics/lhc-run-3-physics-record-energy-starts-tomorrow (2022), accessed: 19-09-2023