Migration of CADI to Fence

Muhammad Imran[1], Andreas Pfeiffer[2], Rao Muhammad Atif Shad[1], Joel Closier[2], Athanasios Samantas[2] and Misha Urooj Khan[1] for CMS Collaboration
[1] *National Centre for Physics, Islamabad, Pakistan*
[2] *CERN, Geneva, Switzerland*

Abstract

- **CMS Analysis Database Interface (CADI):** Central tool for managing publications in the CMS experiment at CERN's Large Hadron Collider (LHC).
- **Frontend engine for Glance (FENCE):** Developed by UFRJ-CERN collaboration, FENCE unifies systems by creating interfaces through JSON configuration files.
- **FENCE Redesign:** LHCb modularized FENCE, splitting it into a PHP REST API backend and a VueJS frontend for improved flexibility.
- **Migration to FENCE:** CMS decided to migrate CADI to the modular FENCE system, following the success in ATLAS, LHCb, and Alice.
- **Subsystems:** Migration initially focused on two FENCE subsystems:
  - **Membership:** Manages members, institutes, authorship, and reports.
  - **ALCM:** Manages workflows such as CADI, facilitating the publication process.
- **Challenges:** Several issues were encountered during the migration process.

Introduction

- CADI is a management tool for physics publications in the CMS experiment.
- Acts as a central database, tracking analysis papers conducted by CMS researchers.
- The journey from early analysis to publication involves multiple stakeholders providing feedback and approvals.

Motivation

- FENCE Originally used by the ATLAS experiment, now has been adopted by LHCb and Alice, utilizing a modular architecture with PHP-based REST API backend and VueJS frontend.
- CMS plans to migrate to the FENCE system, starting with the membership and ALCM subsystems.
- NCP took institutional responsibility for leading the migration of CADI to Fence. The plan involved NCP personnel rotating to CERN for on-site collaboration, working closely with the iCMS and LHCb teams, while the remaining team members provided remote support.

Comparison of Architectures

- A comparison of the architectures of the old CADI and new FENCE systems.

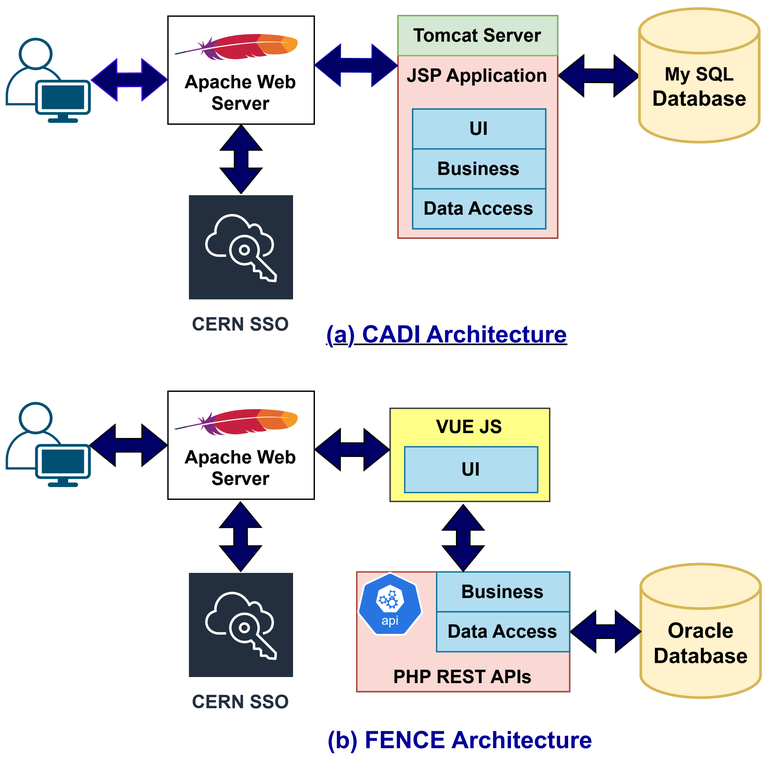

Figure 1. CADI vs. FENCE architecture comparison

Membership

- **Membership** is a prerequisite for all other applications to function.
- It holds essential information such as members, institutes, authorship, and reports

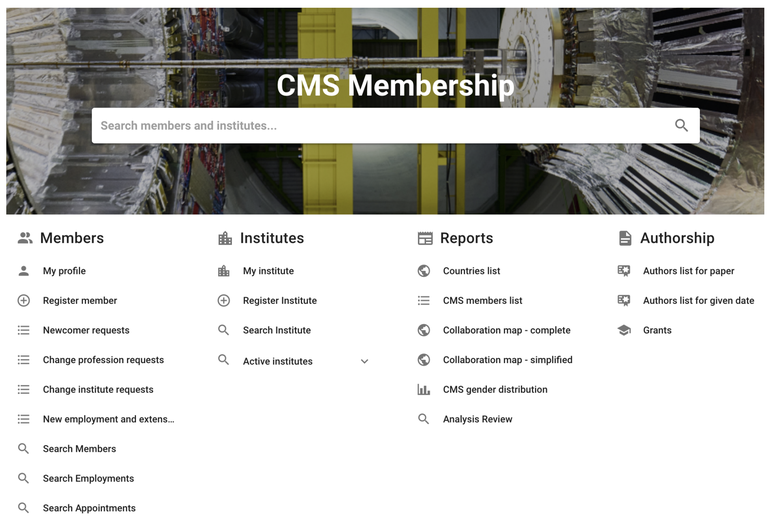

Figure 2. Membership

- This subsystem forms the foundation for other dependent applications such as **ALCM**.

ALCM

The ALCM module manages publications like CADI and includes the following sub-modules:

- **Figures:** Manages the figures/plots associated with publications.
- **Analysis:** Handles the analysis aspects of the publication life-cycle.
- **Notes:** Provides functionality for managing notes related to the publications.

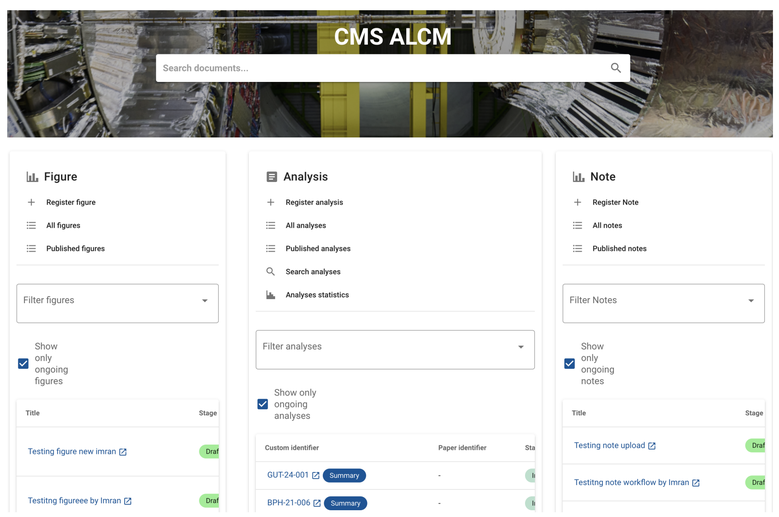

Figure 3. ALCM

- The LHCb ALCM version includes two modules: **Figures** and **Analysis**.
- A new **Notes** module is being introduced in CMS for handling CMS Notes, while Figures and Analysis workflows are being updated to meet CMS requirements based on feedback.

Migration Process

The migration process involves two critical aspects:

- **Data Migration:** Transferring and validating CMS data to fit the FENCE schema.
- **Business Logic Adaptation:** Updating workflows and rules to meet CMS-specific needs within the FENCE framework.

Figure 4. Data Migration

- These are key for a seamless migration, ensuring data integration and functional alignment with CMS.

Issues Faced

The migration encountered several challenges:

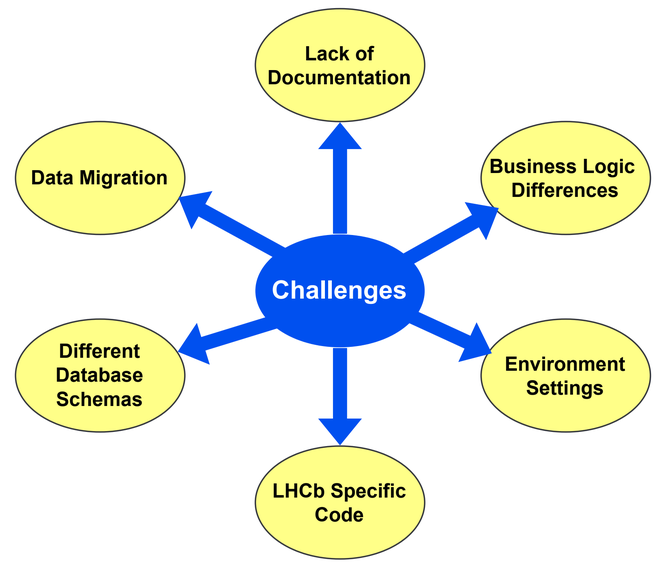

Figure 5. Issues Faced During Migration

Lessons Learned

Key takeaways from the migration process include:

- **Schema Understanding:** Thorough schema analysis is essential for accurate data mapping/importing.
- **Importance of Documentation:** Clear documentation eases implementation and troubleshooting.
- **Stakeholder Engagement:** Regular feedback from stakeholders is crucial for meeting requirements.
- **Environment Consistency:** Consistent settings across environments minimize deployment issues.
- **Testing is Crucial:** Rigorous testing of migration scripts ensures data integrity.
- **Flexible Coding:** Designing adaptable code facilitates future migrations.
- **Plan for Manual Tasks:** Anticipating manual interventions can smooth transitions.

Conclusion & Future Work

- The successful migration of CADI to the FENCE system will enhance publication management and collaboration within CMS.

Future work will include:

- Adapting Membership logic as needed.
- Adjusting workflows based on CMS feedback.
- Evaluating necessary migrations from other modules of "iCMS" (e.g., EPR, awards, jobs, nominations, etc.).

October 19 - 25, 2024
CHEP 2024
Conference on Computing in High Energy and Nuclear Physics