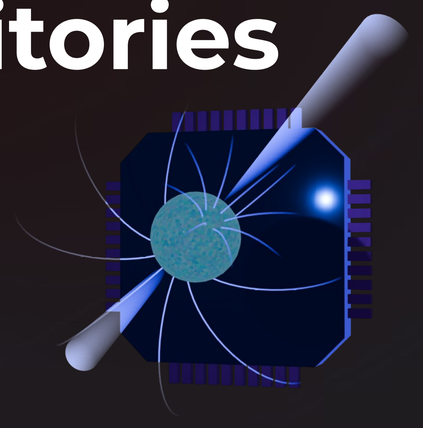
Astronomical Projects with Multiple Repositories

A Portal for Consolidated Information and Kubernetes-Based Operations

Elsa Buchholz, DZA Görlitz, Email: elsa.buchholz@dzastro.de
Co-contributors: Sagar Bora (DZA Görlitz), Marc Drobek (DZA Görlitz), Hermann Heßling (DZA Görlitz)



The antennas and telescopes of the Square Kilometre Array Observatory (SKAO) will produce enormous data streams, making long-term storage of all raw telescope data impractical. To cope with this amount of data, data must be processed in real time and filtered as close to the telescopes as possible, reducing the volume of data that needs to be stored.

Machine Learning-based Pipeline for Pulsar Analysis (ML-PPA) aims to enable automated, real-time pulsar searches during data acquisition. Supporting this approach requires not only an efficient analysis pipeline but also a maintainable software infrastructure that facilitates automated documentation, deployment, and centralized access to project information.

>> ML-PPA: Concept, Architecture and Software

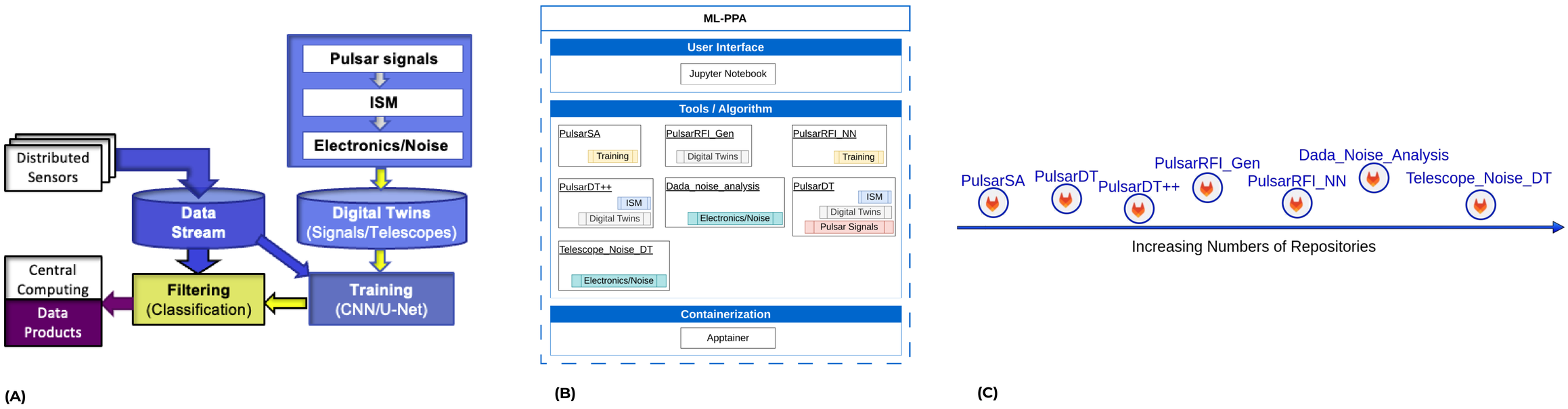


Figure 1: ML-PPA System Architecture and Implementations

The goal of the ML-PPA software framework is to provide a fully automated pipeline for pulsar search. The conceptual structure of the current ML-PPA approach **(A)**, the implementation of the project's three-layer software architecture **(B)** and the growing number of associated software repositories **(C)** are illustrated in the figures.

As the number of software repositories and developers grows, code maintenance and the provision of documentation are becoming increasingly challenging. To address this, measures are being taken to automatically generate and provide documentation, and to consolidate information from various sources in a single location.

>> ML-PPA: Deployment, Operation and Information Portal

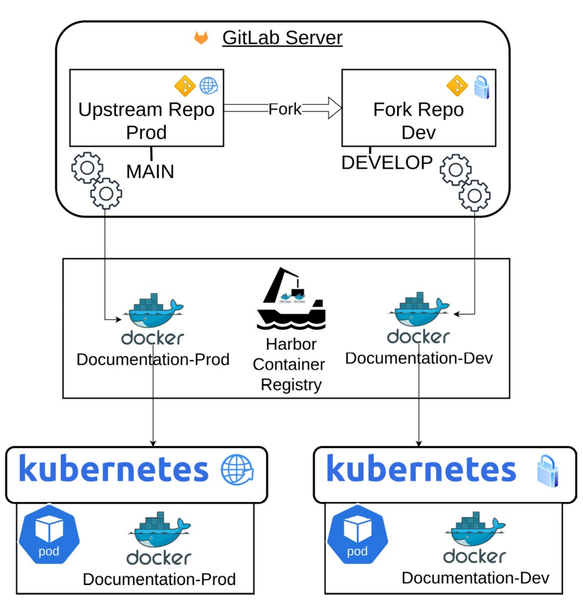


Figure 2: Multi-Environment Documentation Deployment

Separate CI/CD pipelines for the development and production repositories build documentation from source code, package it into Docker images, and publish them to the Harbor container registry. The images are deployed on separate Kubernetes clusters: an internally accessible development cluster and a publicly accessible production cluster.

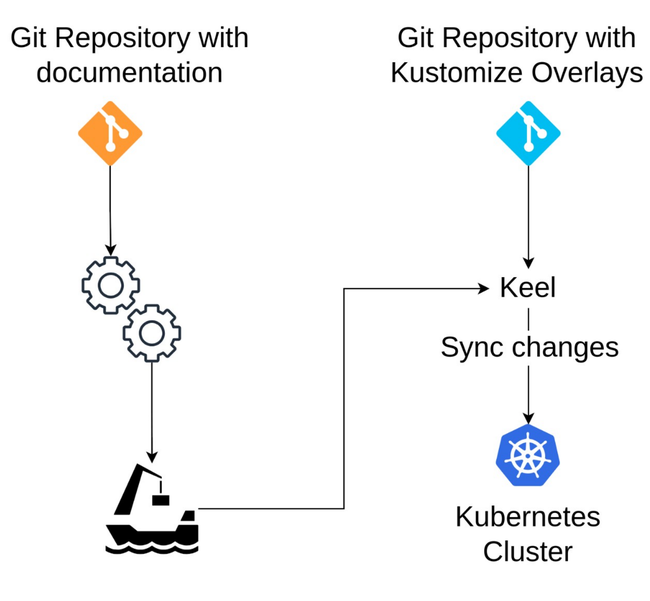


Figure 3: Fully Automated Documentation Updates

Two Git repositories separate tool development from Kubernetes infrastructure management. Keel monitors the documentation images and automatically triggers redeployment when an image update is detected, ensuring that the latest documentation is deployed to the Kubernetes cluster without manual intervention.

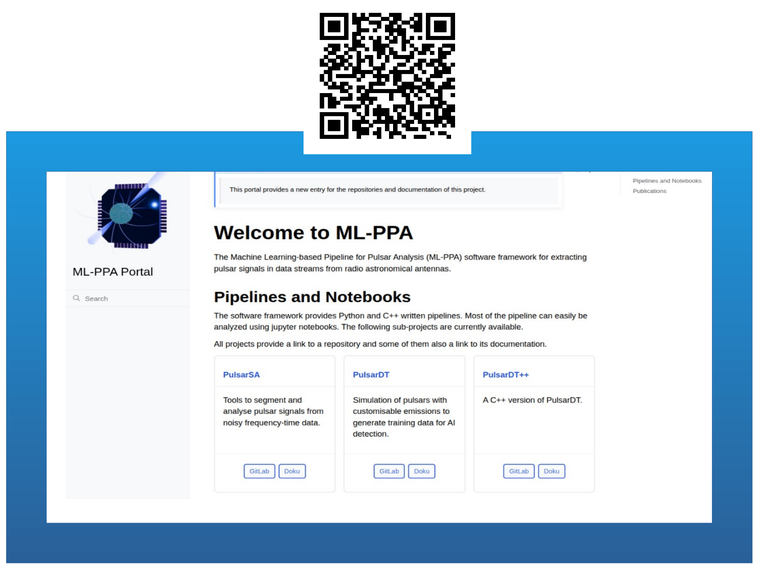


Figure 4: ML-PPA – A Central Information Hub

The ML-PPA portal provides a central access point to project information, linking the documentation and GitLab repositories of individual tools. A QR code enables direct access to the portal.

Acknowledgments
We are grateful for stimulating discussions with Andrei Kazantsev (MPIfR Bonn), Yurii Pidopryhora ((MPIfR Bonn), Marc Drobek (DZA Görlitz), Sagar Borra (DZA Görlitz), Tanumoy Saha (DZA Görlitz), Marcel Trattner (HTW Berlin), Yanik Yiannakis (MPIfR Bonn), Lars Haupt (DZA Görlitz)

PUNCH 4NFDI

DEUTSCHES ZENTRUM ASTROPHYSIK