Karlsruhe Institute of Technology

Update: Transition to Phoebus at the Karlsruhe Institute of Technology

Edmund Blomley & Sebastian Marsching

Outline

- KIT Accelerators & Environment
- Backend Infrastructure
- User Space
- Panel Migration
- Summary & Outlook

Location: Karlsruhe – Germany

Karlsruhe Research Accelerator (KARA)

- KIT synchrotron light source & accelerator test facility
  - until 2015 known as “ANKA”
- Circumference: 110.4 m
- Ramped storage ring: 0.5 - 2.5 GeV
- **Mostly “stable“ IOCs & panels**

- One week per month:
  - Beam physics experiments and tests
  - Very variable in energy, filing pattern, bunch current and operational condition

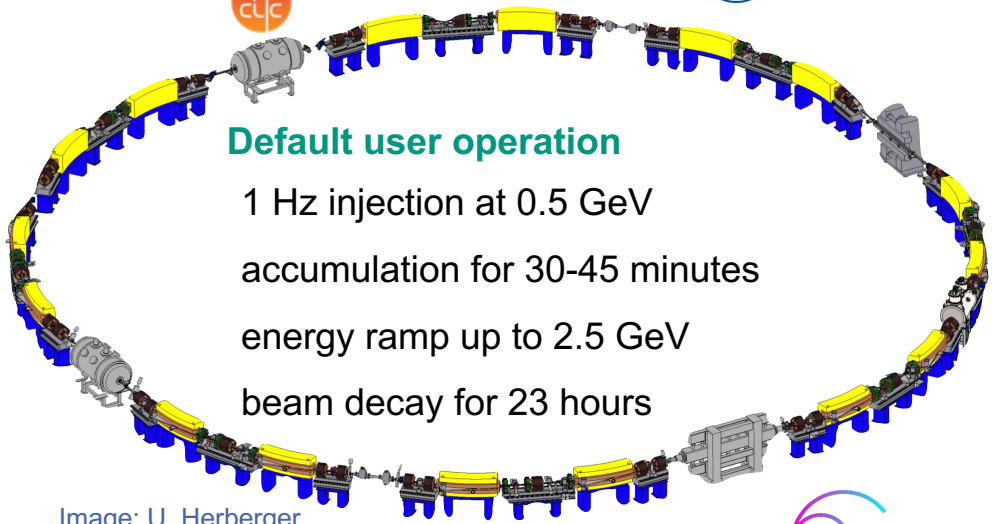
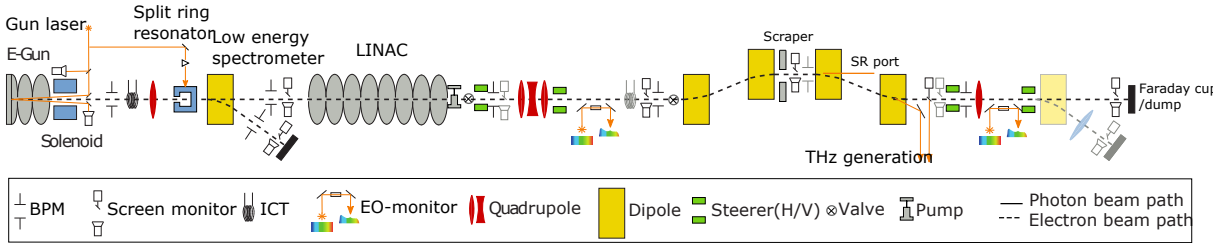


Image: U. Herberger

Far-Infrared Linac and Test Experiment

- 50-90 MeV linear electron accelerator
- Goal: Ultra short electron pulses (1-300 fs), THz experiments

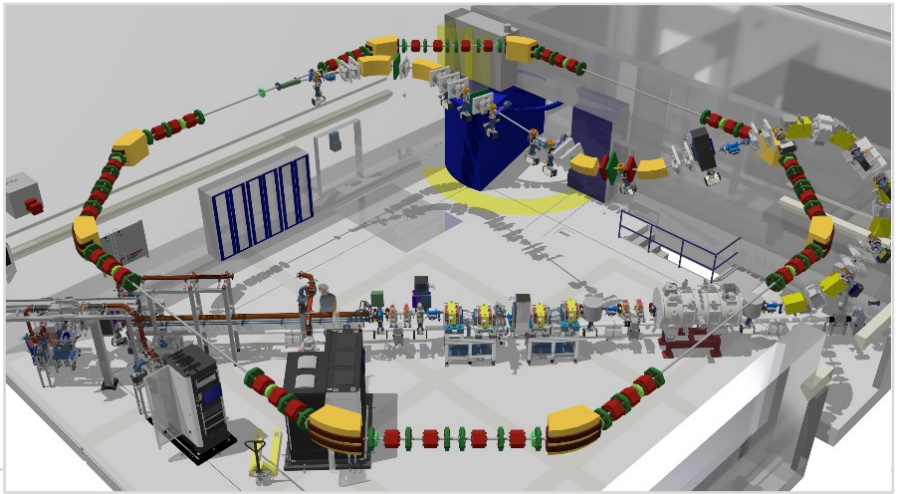
- **Panels and IOCs much more fluid**
  - Overlap in panels, IOC support and general controls infrastructure
  - But separate network, building, operators and users



Very Large Acceptance compact Storage Ring (VLA-cSR)

- New storage ring in TDR/FDR phase
- Circumference: 43.2 m (6.9 MHz revolution frequency)
- Installation and commissioning 2026-2027
- Store ultra-short bunches (~ 10 fs – 1 ps)
  - from FLUTE
  - from LPA(s)

- **No panels or IOCs yet**
  - Extension to FLUTE controls
  - Opportunity to rework/redesign panels



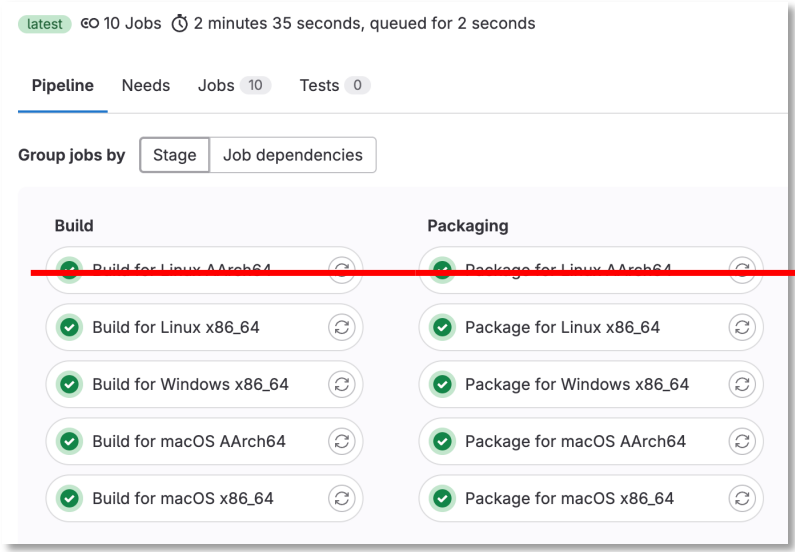
Controls Environment

- **EPICS 7**
  - Still 99% Channel Access
  - IOCs mostly on virtual machines
  - GUI: Control System Studio
- **Ubuntu LTS** for all servers and terminals
  - About to migrate to 24.04
- Code repository with lots of CI usage: **GitLab**
  - IOCs, EPICS distribution, software packaging, …
- IT orchestration: **Salt Project**
  - (comparable to Ansible)

Backend Infrastructure

Building Phoebus

- Try to be as close to upstream as possible
- GitLab CI to build for all platforms
  - Using only Linux Docker images
  - (Removed Linux ARM build due to issues with JavaFX)
- Modifications made to upstream
  - **EPICS Jackie** PV source (alternative to JCA / CAJ)
  - JSON / HTTP based archive reader
  - Improvements to Elasticsearch and Kafka connection code



Alarm Server & Logger

- Each build separately to its own Docker image
- Alarm server container running in “host” network mode
- Current issue for alarm server: experiencing long load times during channel import
- Using Elasticsearch and Kafka HA clusters (changes to connection code have been merged upstream)

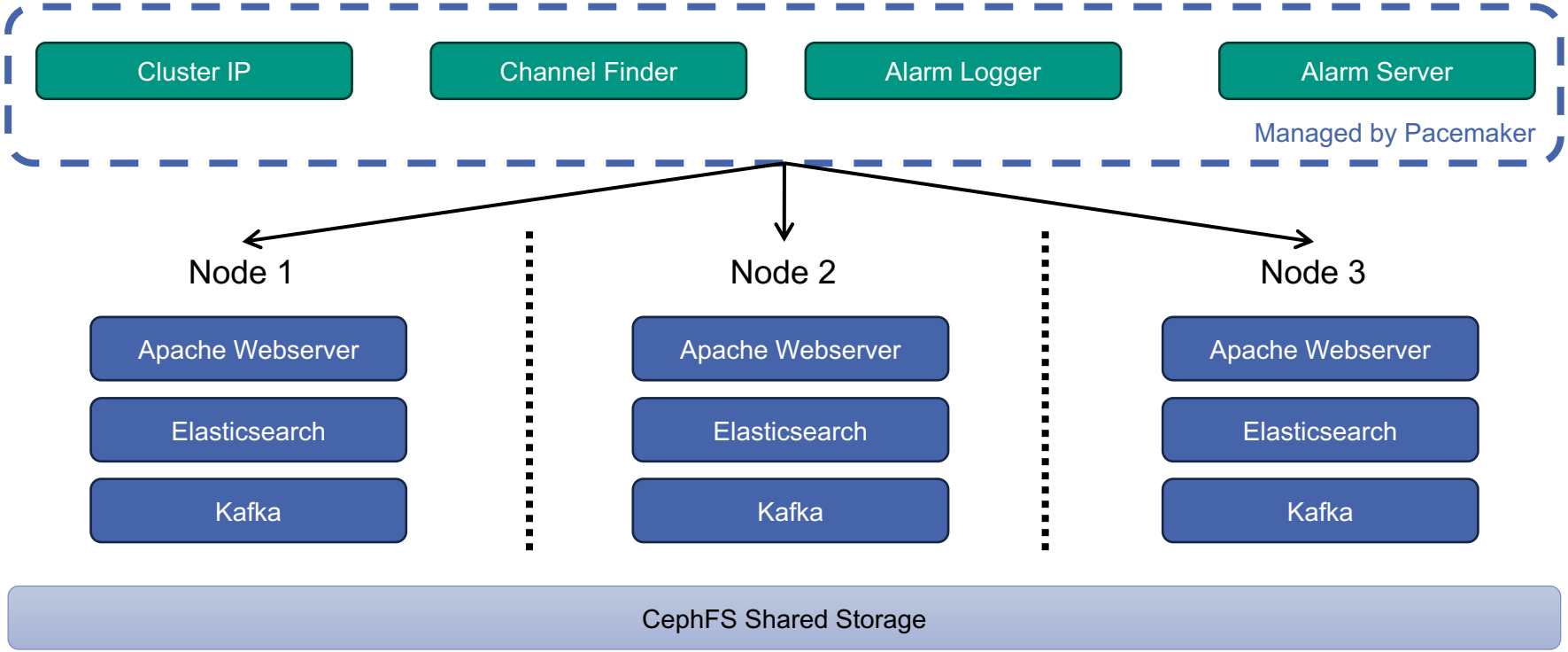
Channel Finder

- Use existing PV export feature to feed channel finder server
  - Was used so far for PV bash completion
  - No recast modules in IOCs
- Might change at a later point of time to replace custom PV export
- Improvements to Elasticsearch connection code:
  - Allow specifying multiple servers (HA cluster)
  - Allow use of HTTPS
  - Allow authentication

Phoebus Server Cluster

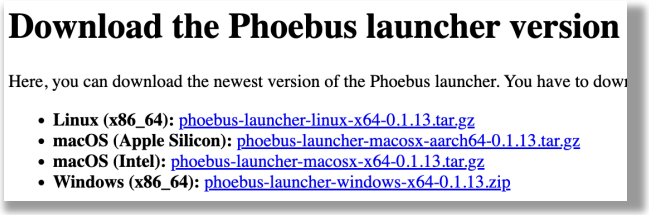
- Manage HA services via **Pacemaker**
- Node setup via **Salt**
- File storage via **CephFS** cluster
- **Elasticsearch** and **Kafka** running as a three-node HA cluster
- Clients download Phoebus client, configuration, and panels from web server using a **HA IP address**.

Phoebus Server Cluster



User Space

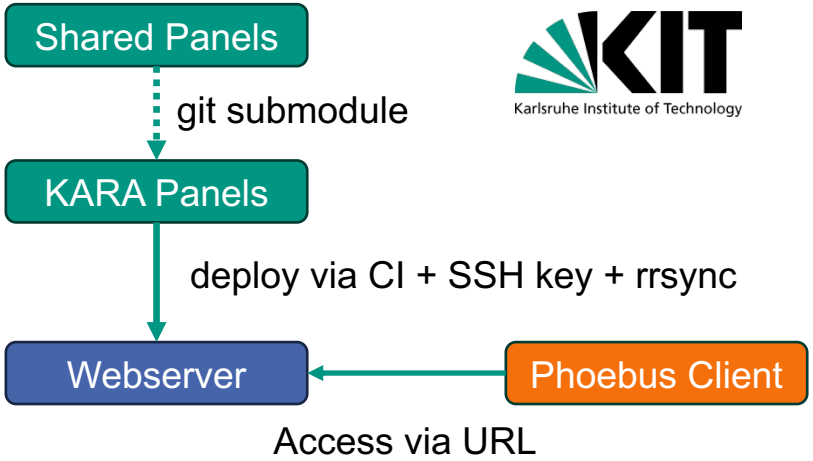
Phoebus Distribution



Karlsruhe Institute of Technology

- Download and installation via **webserver**
  - **phoebus.kara.ibpt.kit.edu**
  - Based on client network decides for **office** or **accelerator** version
- Custom launcher
  - Updates Phoebus version and configuration settings automatically
  - `settings.ini` & `kafka.properties`
- Provides environments
  - Choose from stored mementos (layouts)

Panel Distribution



- Panels are stored in GitLab
  - One project per accelerator
  - CI pipeline deploys panels to webserver
- Shared project
  - EPICS modules (areaDetector)
  - IOCs used across both accelerators
  - Common “resource” panels

alternative?

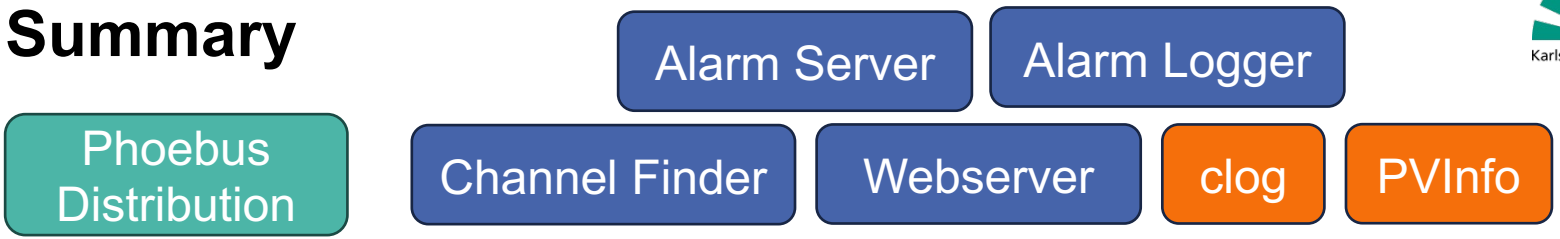
- By default, panels (and mementos) are accessed via the webserver
- Path to (local) panels can be provided to launcher
  - Mostly for panel development

Panel Migration

- Phoebus can open .opi files and migrate to .bob files
  - **AdvancedConvertor** for mass migration
- ~10 years of CSS panels across two accelerators
  - Created and maintained by different groups
- Decided for “manual” migration approach
  - Includes re-structuring and clean-up of old panels
  - Re-align style and behaviour for both accelerators

  → **Much slower migration, but hopefully worth the effort**

Summary



- High availability: Phoebus backend services with docker in cluster setup
- Simple end-user interface
  - Only needs to download one file once
  - Self-configurating & automatically updating
  - Remote panels
- Use “opportunity” to re-work/structure long history of panel development
- Next steps: new electronic logbook & PVInfo