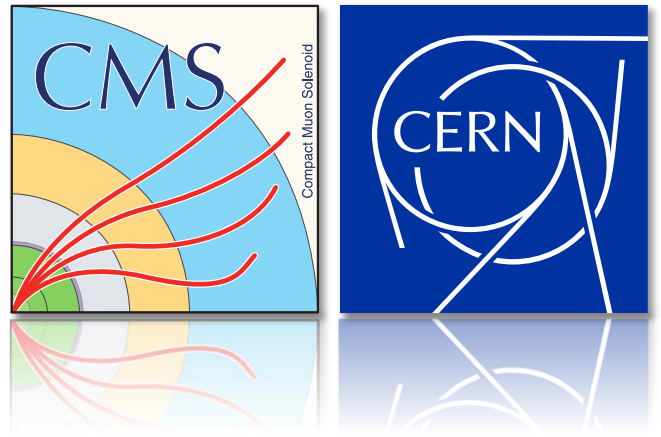


Architecting software applications in containerized environment for CMS data acquisition

Presenter: Dainius Šimelevičius on behalf of CMS DAQ group
Primary authors: Luciano Orsini - CERN, Dainius Šimelevičius - CERN/Vilnius University

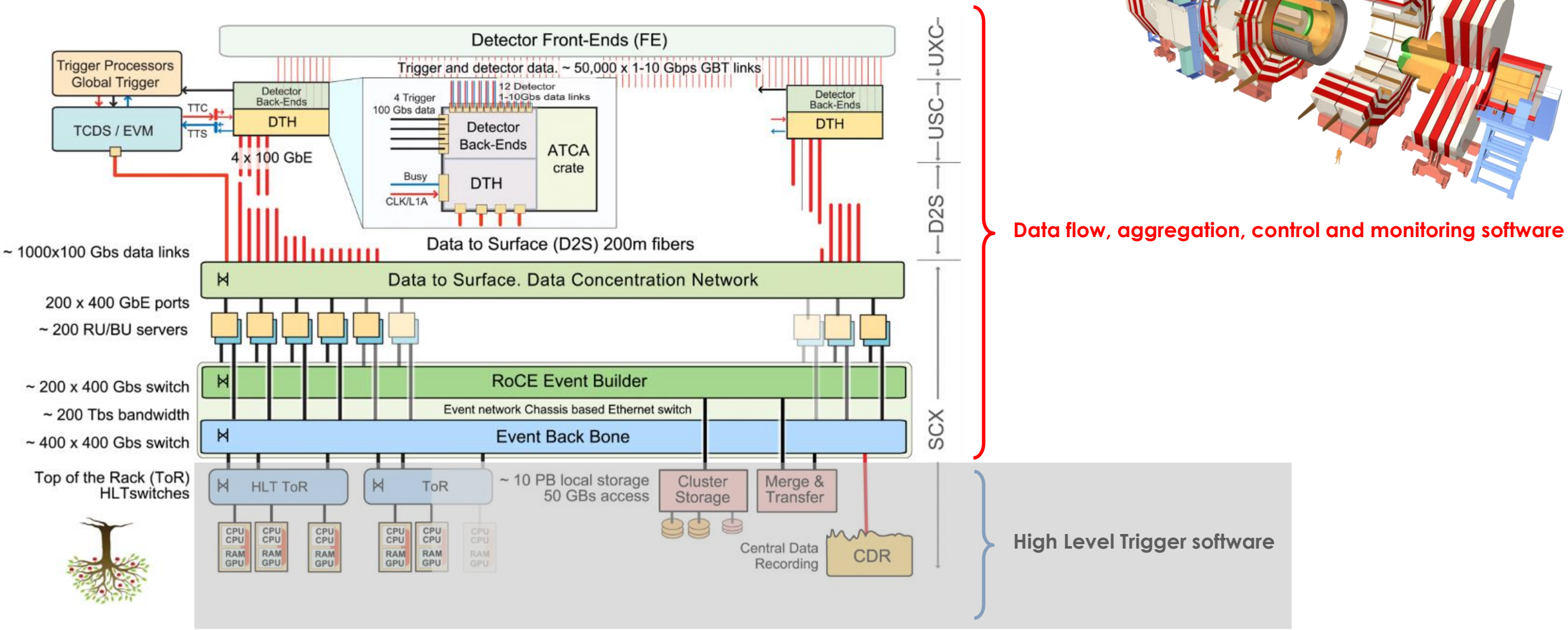
CHEP 2024
27th International Conference on Computing in High Energy and Nuclear Physics
October 19-25, 2024, Kraków, Poland

Outline

- Introduction
- Motivation
- Architecture
- Technologies
- Software release
- Conclusion

Introduction

CMS Data Acquisition (DAQ) for HL-LHC (Phase 2)



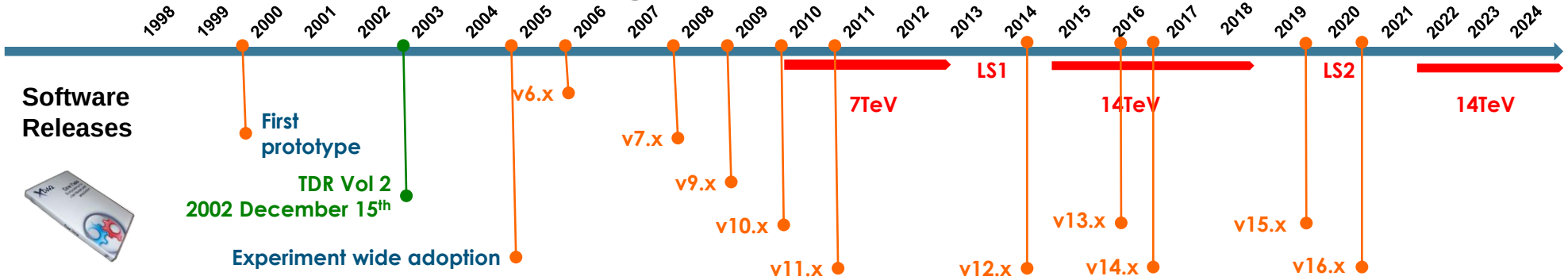
DAQ system and software

- DAQ runs on a large computer cluster with a high-performance network interconnect, and it has demanding requirements on control, monitoring and configuration
- DAQ system must accommodate various application scenarios, such as interfacing with external systems, accessing custom electronics devices for data readout, and event building
- DAQ is implemented as a service-oriented architecture where DAQ applications, as well as general applications such as monitoring and error reporting, are run as self-contained services
- DAQ software is implemented using C++ software framework called XDAQ

Motivation

Software framework

- A framework (XDAQ) is used in CMS DAQ for many years and is a mature, reliable software
- Pre-designed approach for data acquisition application development
- Uniform approach to application configuration
- Reusable components for application developers
- Components are well tested due to diverse user community
- The framework is being upgraded to provide uniform APIs for the optional usage of several off-the-shelf technologies
- Modernizing framework to be suitable for containerization and orchestration environments and to benefit from new technologies



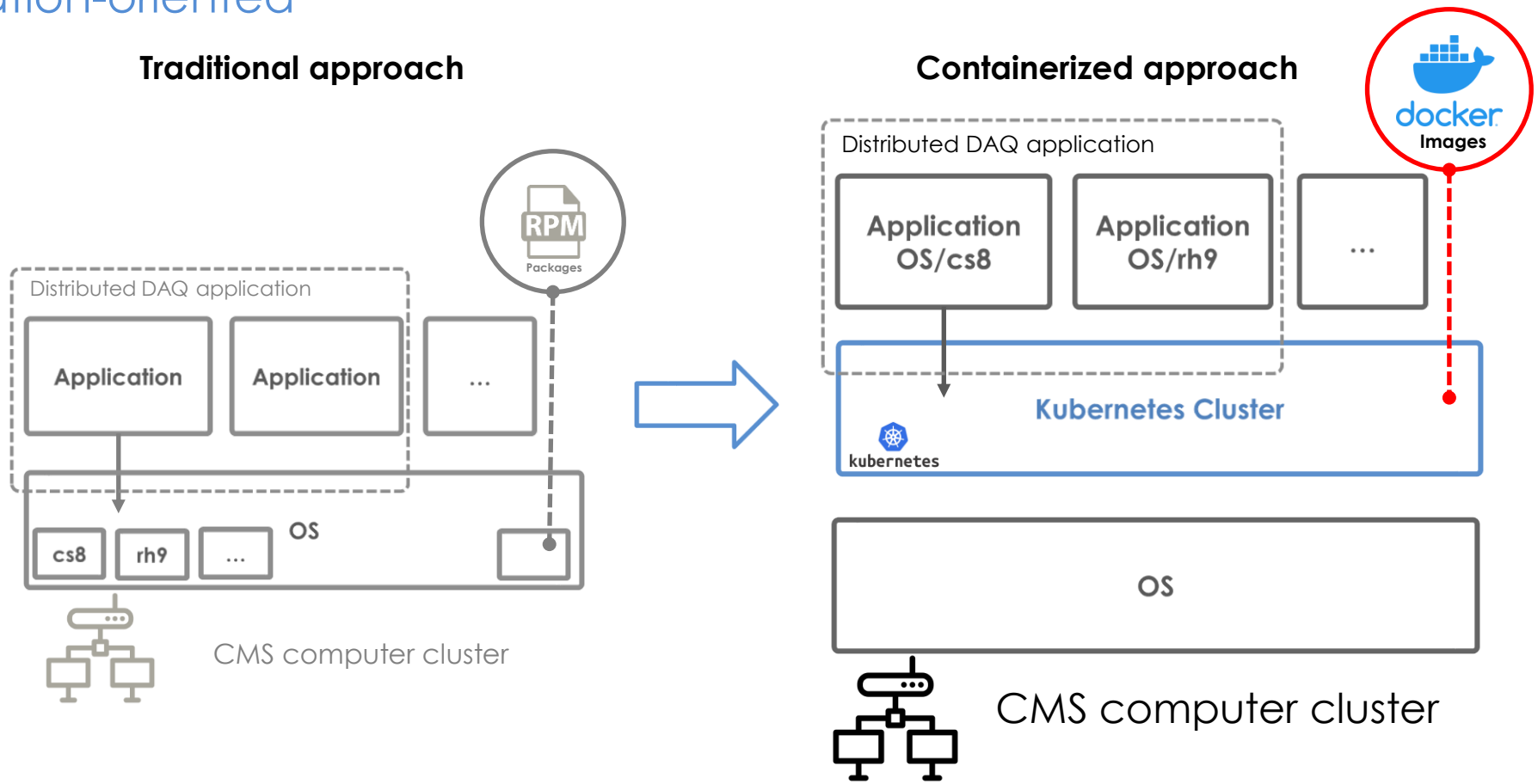
Motivation for containerization

- Software containerization allows restructuring the system into a homogeneous architecture, where all applications are orchestrated in an environment with standardized facilities
- More reliable validation of software in test or production environments
  - Consistent installation on different targets is guaranteed by image construction at build time
  - Tests in production environment do not disrupt installation, roll back of installed software is not needed
- Container image encapsulates all (or almost all) of an application's dependencies
- Containerization helps to avoid infrastructure lock-in caused by dependency on a given operating system and libraries

Architecture

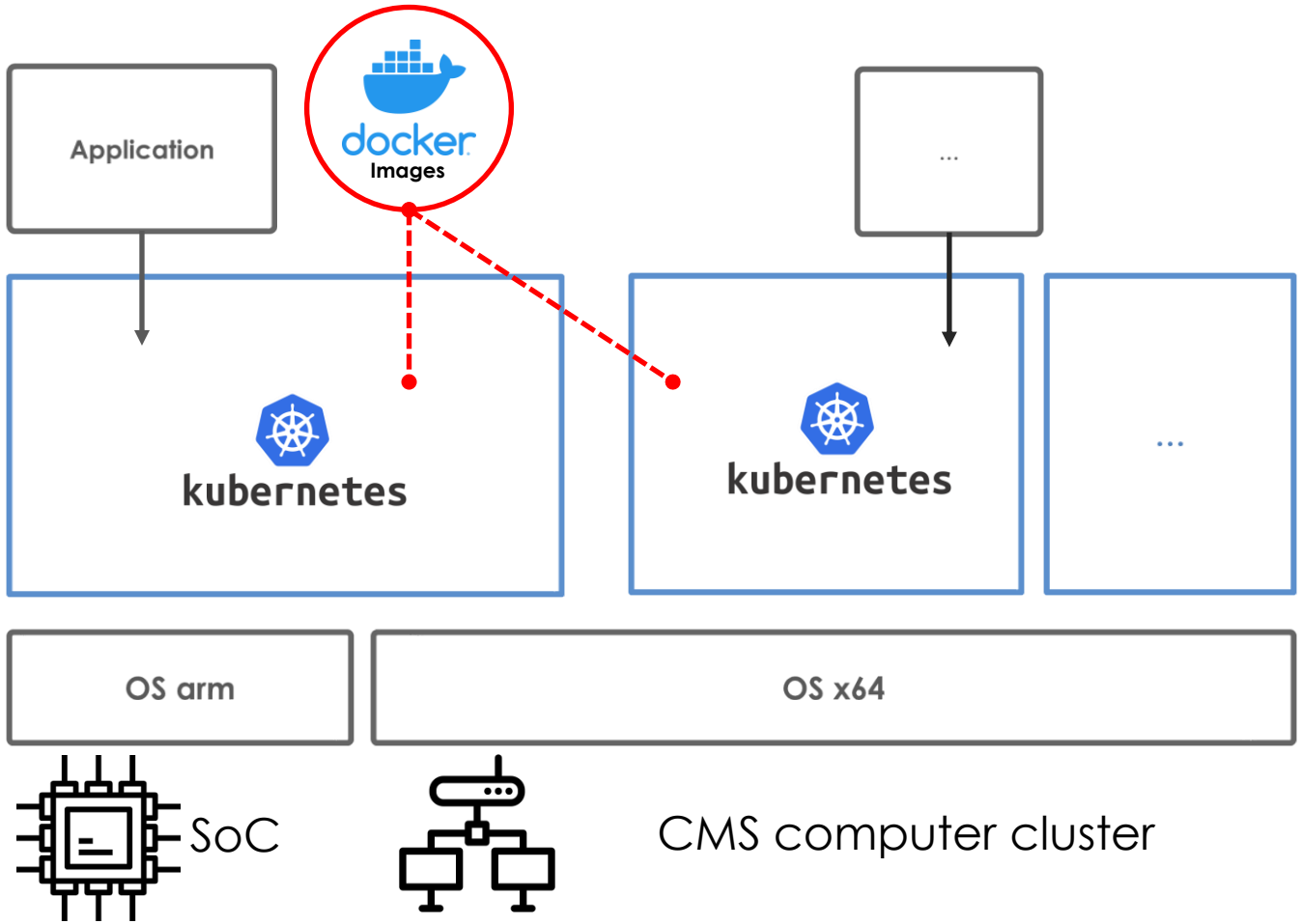
Containerization approach

- DAQ application is a software component designed to carry out a specific DAQ task (e.g., event building, monitoring, error reporting)
- Containerization transforms a software infrastructure from being machine-oriented to being application-oriented



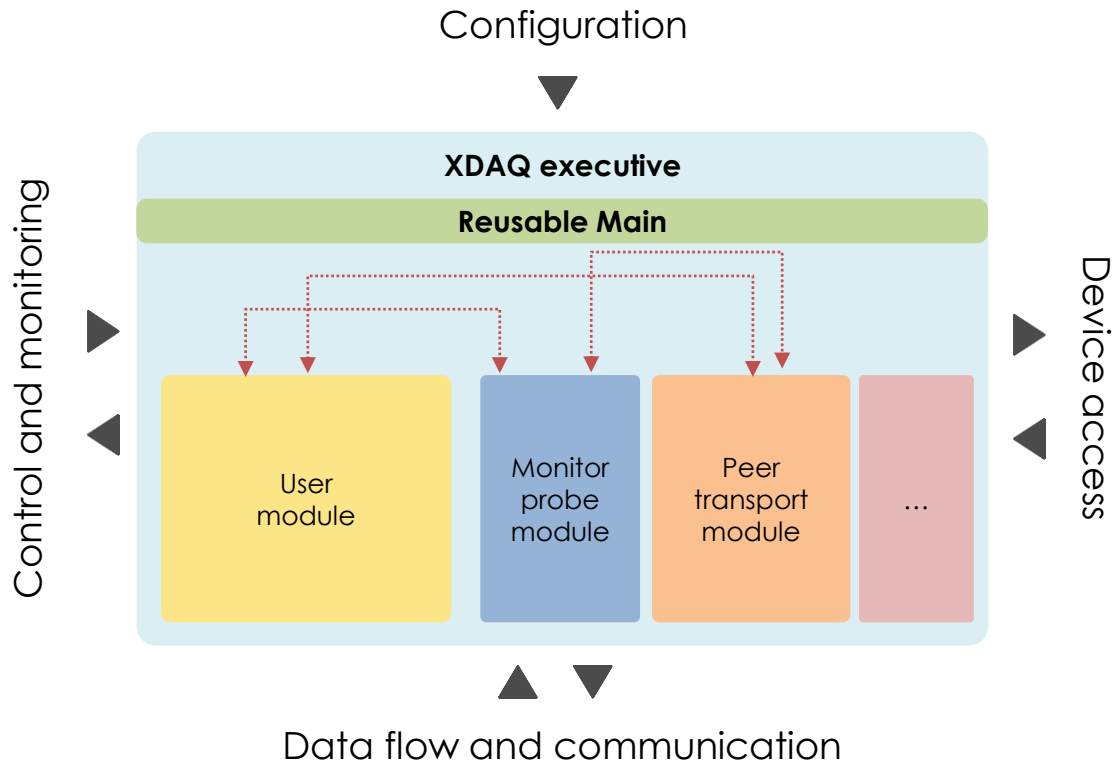
Partitioning computer cluster into K8s clusters

- Several Kubernetes clusters for different purposes, e.g., central DAQ, development cluster, sub-detector clusters
- Dedicated resource allocation and protection



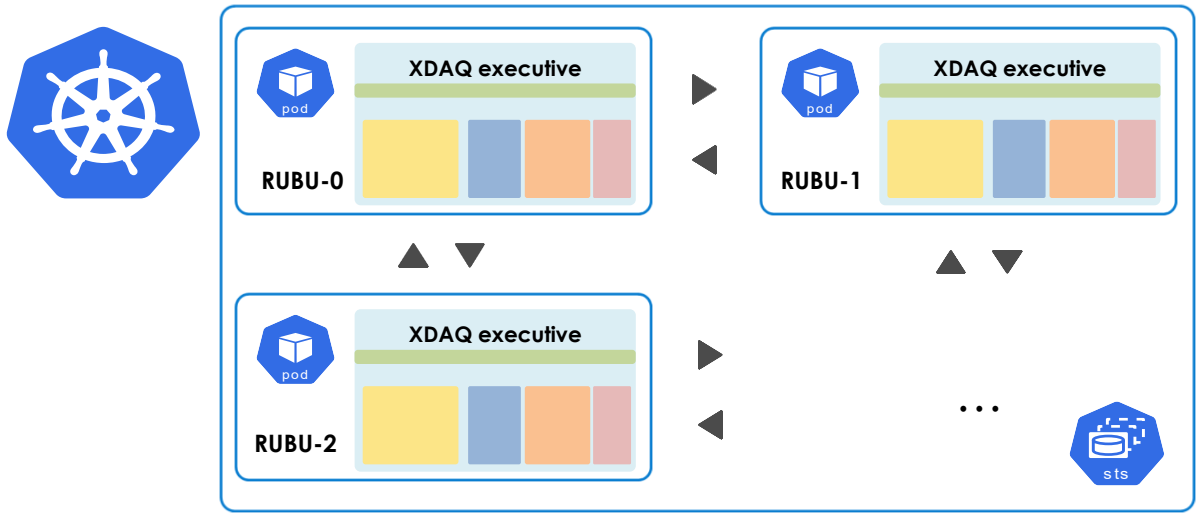
Fundamentals of application architecture

- XDAQ provides a uniform design pattern for implementing DAQ applications
- The main building block of an application implemented with XDAQ is a XDAQ executive
  - Executive corresponds to a Linux executable program
  - Executive can host several XDAQ modules which can be custom or reusable
- Framework provides interfaces for configuration, control and monitoring, communication and device access



XDAQ executives in Kubernetes

- DAQ applications are highly inter-communicating distributed applications
- Containerized XDAQ executives operate within Kubernetes pods, with each pod assigned a unique network identity configured through Kubernetes manifests by the developer. As a result, executives within a Kubernetes cluster maintain the same address, even if physical machine hostnames change or the application is deployed on a different Kubernetes cluster.
- Kubernetes provides convenient tools to replicate copies of the same executive (StatefulSet, Deployment, DaemonSet, etc.)



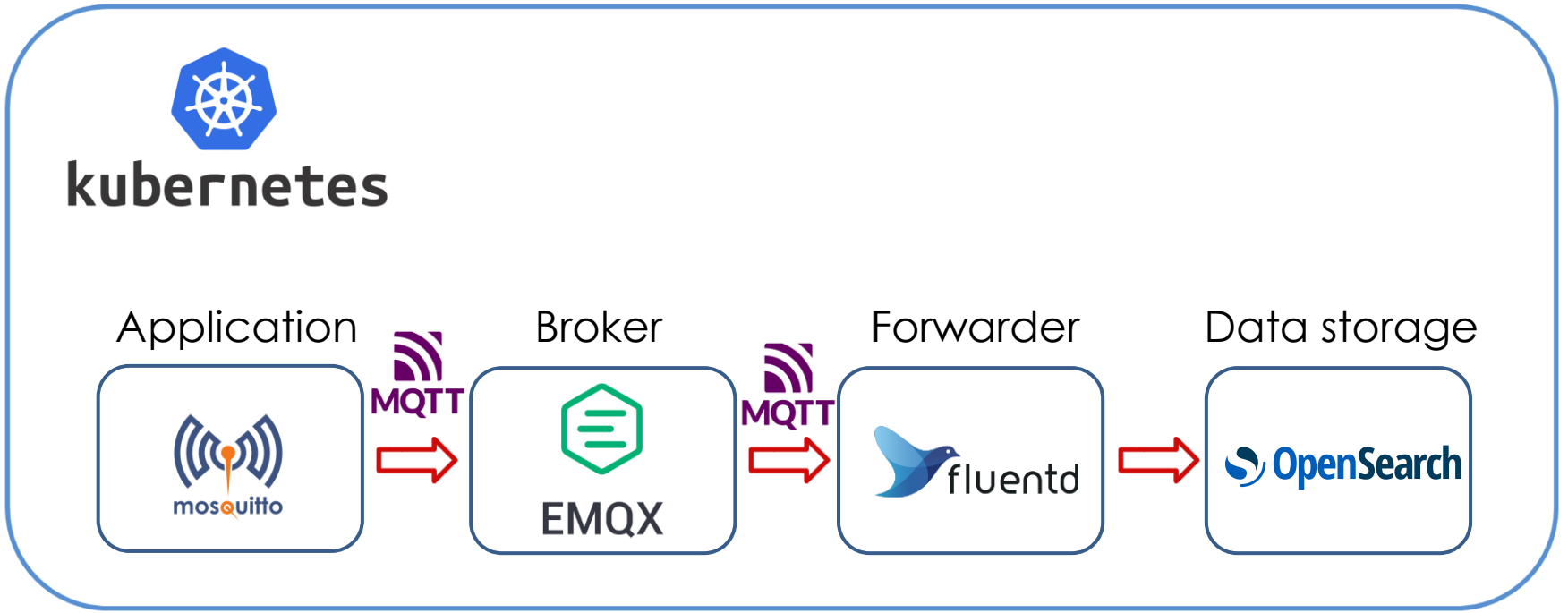
Framework integration into containerized environment

- The same technologies as used in Kubernetes environment
  - HTTP, REST
  - YAML, JSON
- Utilities for interacting with Kubernetes environment
  - K8s API (REST)
  - K8s events
  - Readiness/liveness probes (REST)
  - Networking
  - Web access (through Ingress)
- Support for parameterization of executives running in Kubernetes pods

Technologies

Off-the-shelf technologies for monitoring

- MQTT protocol is used for communication between application and message broker as well as between broker and forwarder
  - Mosquitto library used as MQTT client
  - EMQX used as MQTT message broker
  - Fluentd used as forwarder
  - Opensearch used for data storage
- All components run as containers inside Kubernetes pods



Software release

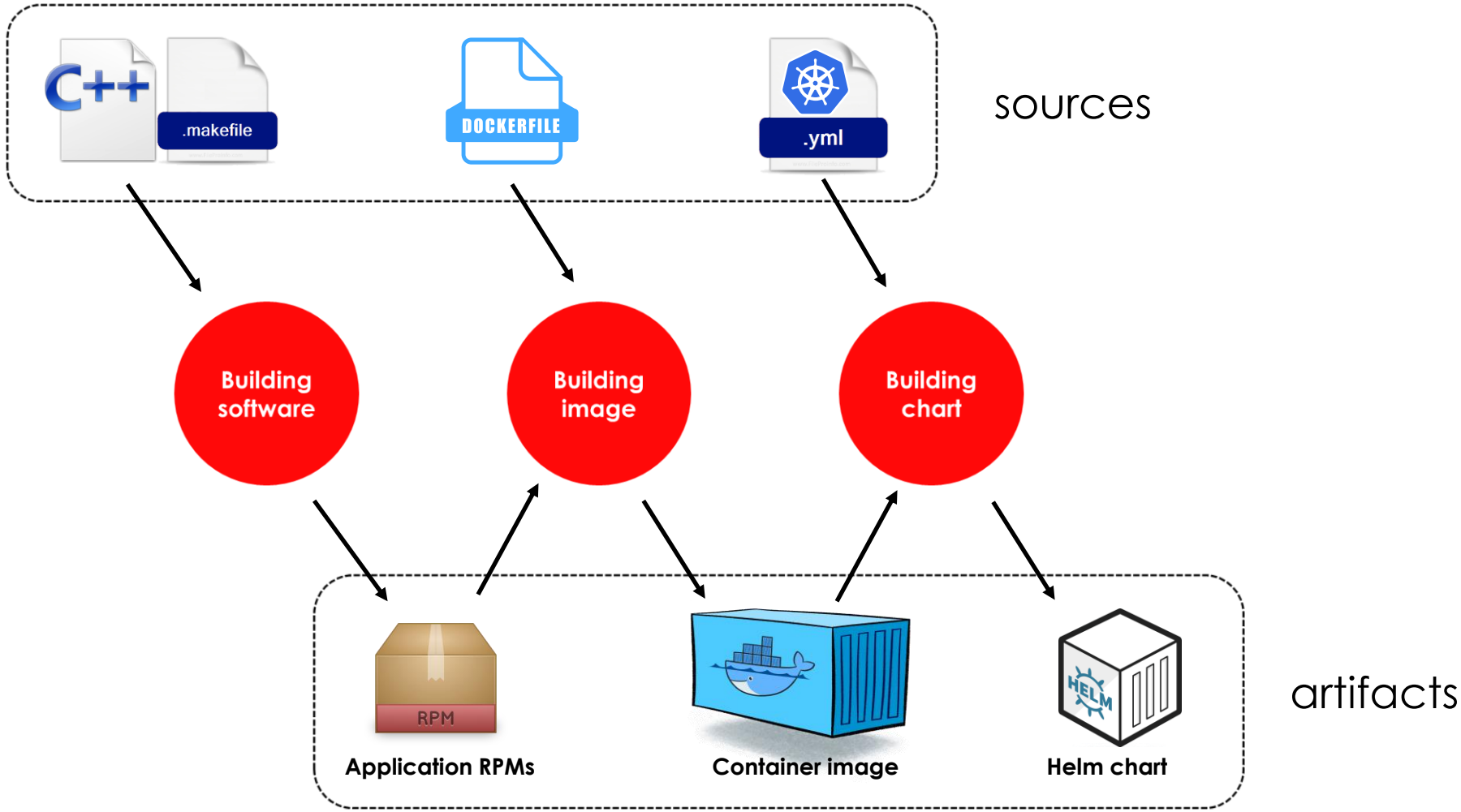
Development to release

- Helm package manager is used for configuring, releasing and deploying containerized software in Kubernetes

Step 1: building and packaging applications

Step 2: containerizing applications

Step 3: managing applications in Kubernetes



Conclusion

Conclusion

- XDAQ software framework is a consolidated and reliable software that can be used during LHC Phase 2
- The framework was modernized to provide a built-in support for containerization and orchestration environments
- Offers tools for creating DAQ applications in a uniform way
- Facilitates application development by offering reusable, well tested components
- Containerization and orchestration address most drawbacks of a traditional infrastructure
- The containerization approach is well suited for DAQ applications, which are generally distributed inter-communicating applications