CMS Data Acquisition System Software

G. Bauer[1], U. Behrens[2], K. Biery[3], J. Branson[4], E. Cano[5], H. Cheung[3], M. Ciganek[5], S. Cittolin[5], J.A. Coarasa[4,5], C. Deldicque[5], E. Dusinberre[4], S. Erhan[5,6], F. Fortes Rodrigues[7], D. Gigi[5], F. Glege[5], R. Gomez-Reino[5], **J. Gutleber**[5], D. Hatton[2], J.F. Laurens[5], J.A. Lopez Perez[5], F. Meijers[5], E. Meschi[5], A. Meyer[2,5], R. Mommsen[3], R. Moser[5,9], V. O'Dell[3], A. Oh[5], L.B. Orsini[5], V. Patras[5], C. Paus[1], A. Petrucci[4], M. Pieri[4], A. Racz[5], H. Sakulin[5], M. Sani[4], P. Schieferdecker[5], C. Schwick[5], D. Shpakov[3], S. Simon[4], K. Sumorok[1], J. Varela[5,8], M. Zanetti[5]

[1]MIT, Cambridge, USA; [2]DESY, Hamburg, Germany; [3]FNAL, Chicago, USA; [4]UCSD, San Diego, USA; [5]**CERN**, Geneva, Switzerland; [6]UCLA, Los Angeles, USA; [7]CEFET/RJ, Brazil; [8]LIP, Lisbon, Portugal; [9]Technical University, Vienna, Austria

CHEP - March, 2009

Outline

- Motivation & Requirements
- Architecture
- Codebase
- History of the development
- Achievements
  - Event builder
  - HyperDAQ
- Summary
- Conclusion

Motivation

- CMS consists of a set of sub-projects
  - Similar to a coordinated set of small experiments
  - Many scenarios: central DAQ, subdetector DAQ, testbeams,
- Geographically dispersed participants
- Autonomous developments
- High personnel turnover
- High performance requirements
- Long lifetime and need to survive technology generations
- Similar tasks to be performed in each sub-detector

Need for integration

No single tool available

Functional Requirements

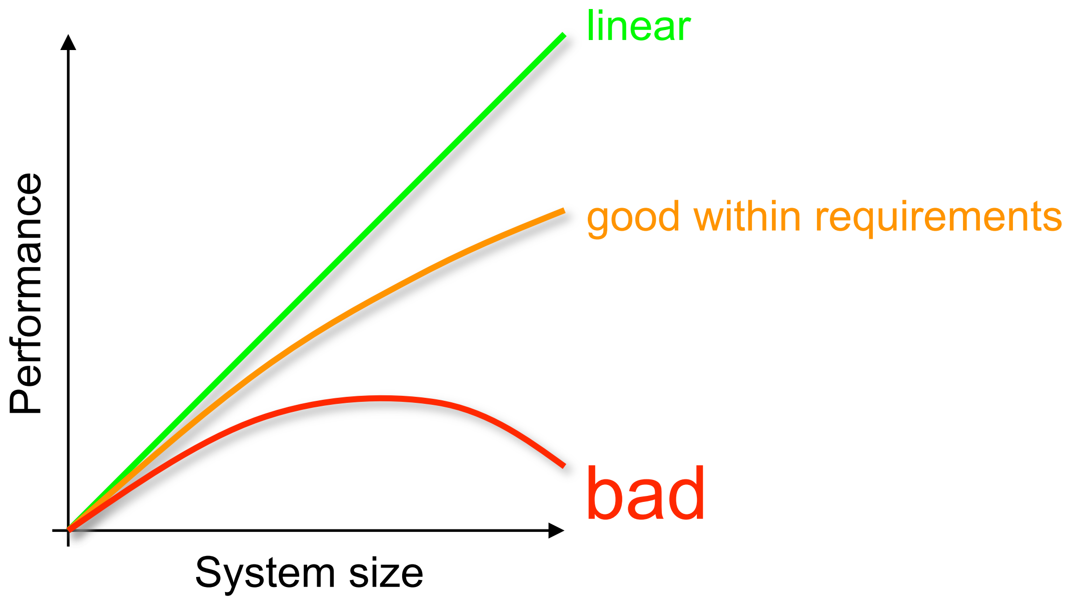
- **Communication and Interoperability**
  - Transparent use of communication protocols
  - Possibility to add new protocols
  - Concurrent use of multiple protocols
- **Device Access**
  - Access to custom devices
  - Transparent access to local & remote devices (bus adapters)
- **Configuration, control and monitoring**
  - Inspect and modify simple/complex parameters
  - Coordinate of application components
  - Record structured information
    - Uniform logging, error reporting, monitoring
    - Interface to persistent data stores

Non-Functional Requirements

- **Maintainability and Portability**
  - Portability across operating system and hardware platforms
  - Add new electronics without functional changes in user software
  - Memory management functionality to
    - improve robustness
    - give room for efficiency improvements
  - Application code shall be invariant with respect to the physical location and the network
  - Foster working with re-usable building blocks

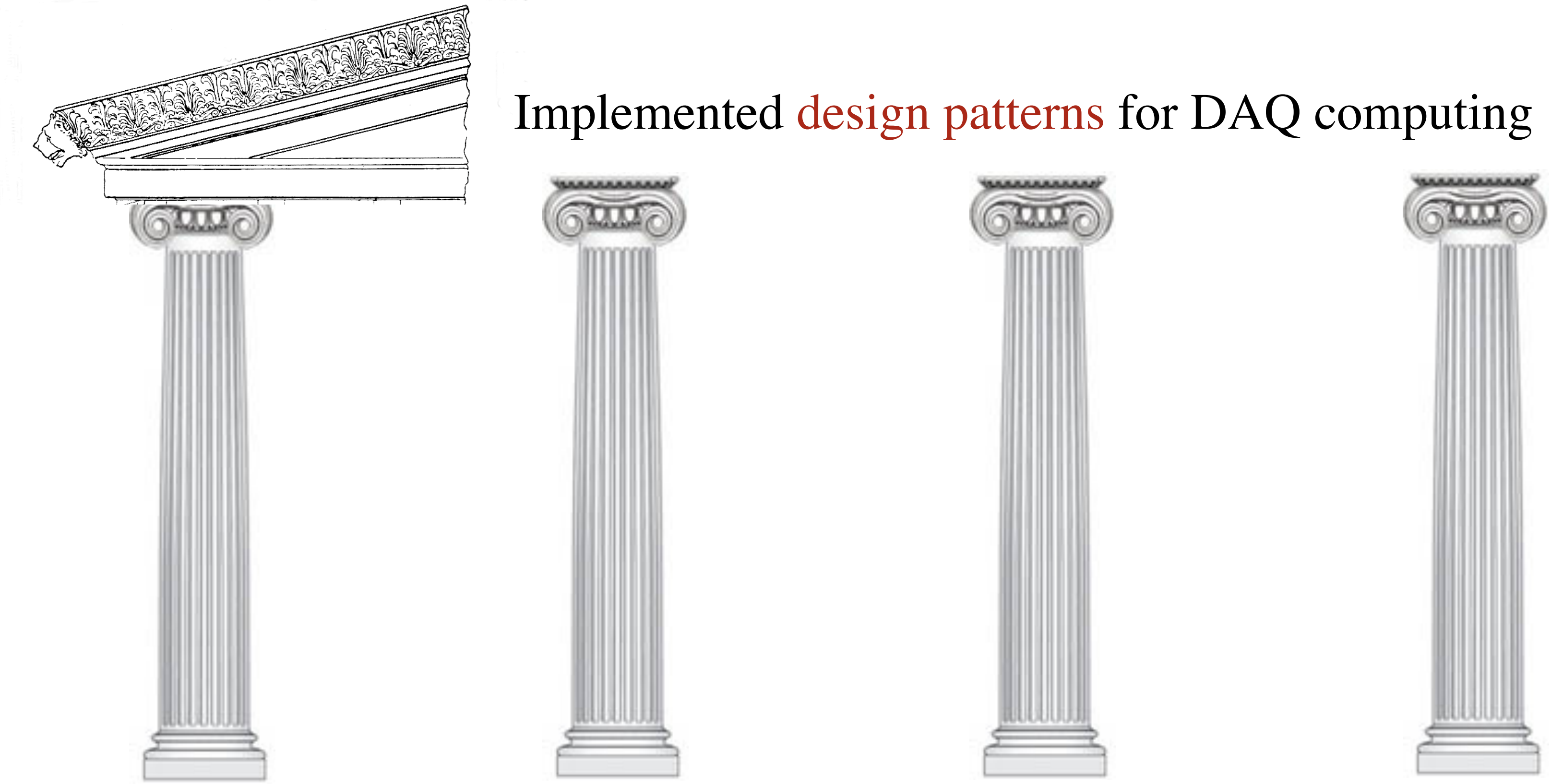
Non-functional Requirements

- **Scalability**
  - Operate within requirements if size or volumes change
  - Take advantage of additional resource availability
  - Overhead introduced by the software environment must be constant for each transmission operation and small with respect to the underlying communication hardware in order not to introduce unpredictable behaviour



Software Infrastructure

- Software as an integral part of the architectural scaffolding
- Software provides centrally designed efficiency enablers



Implemented design patterns for DAQ computing

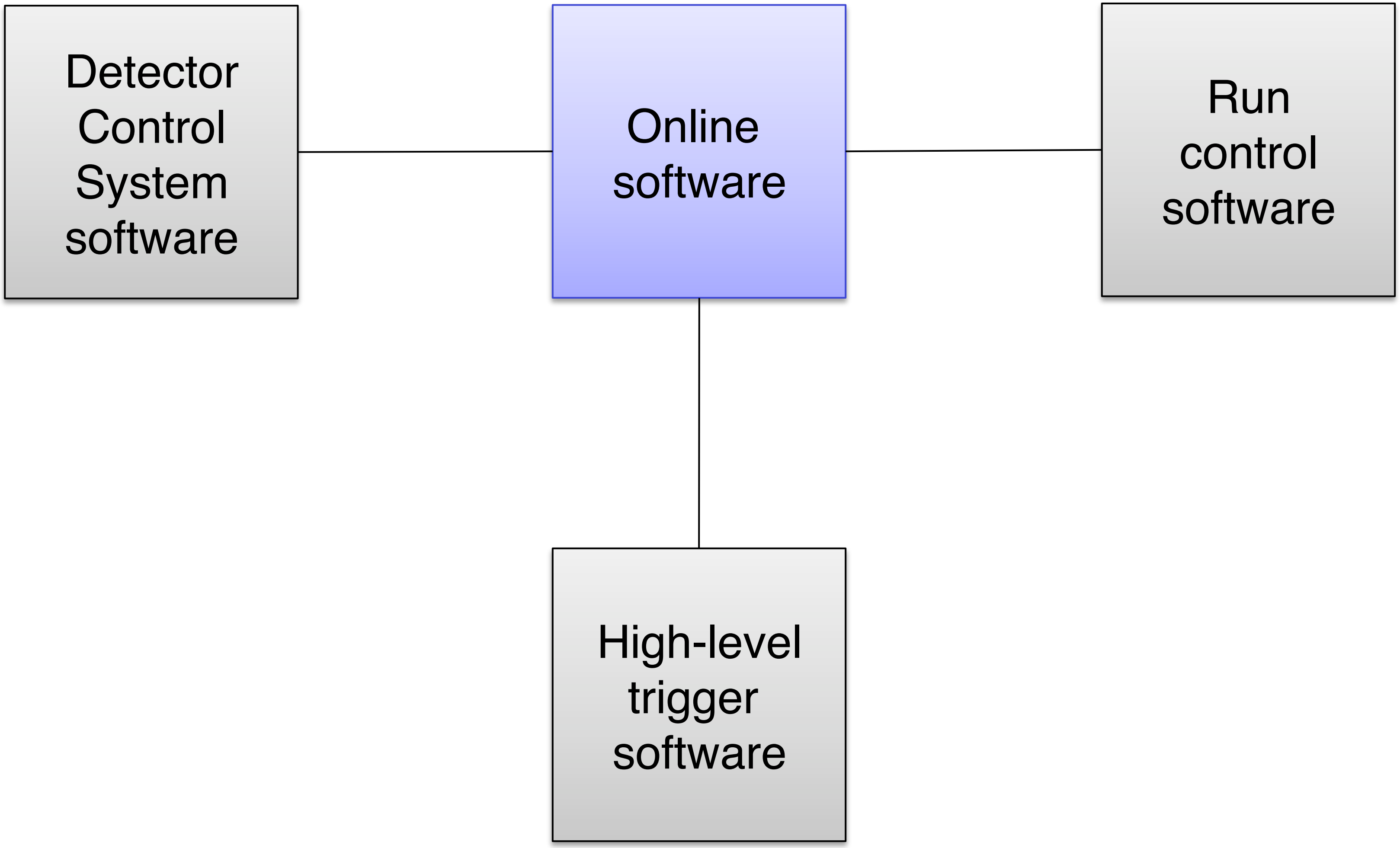
Uniform Addressing

Uniform Communication

Uniform Configuration

Uniform Monitoring & Error processing

Context



Detector Control System software

Online software

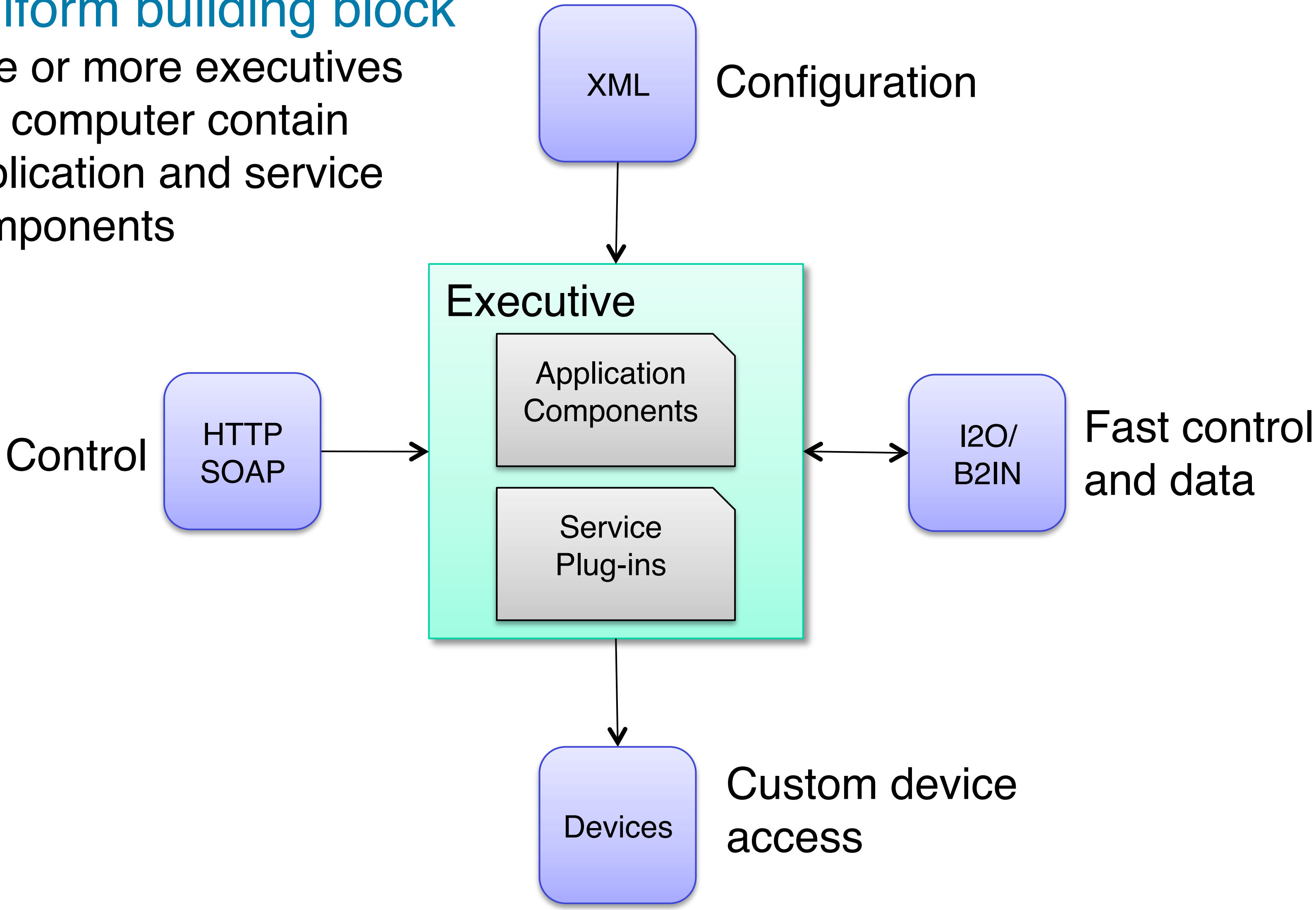
Run control software

High-level trigger software

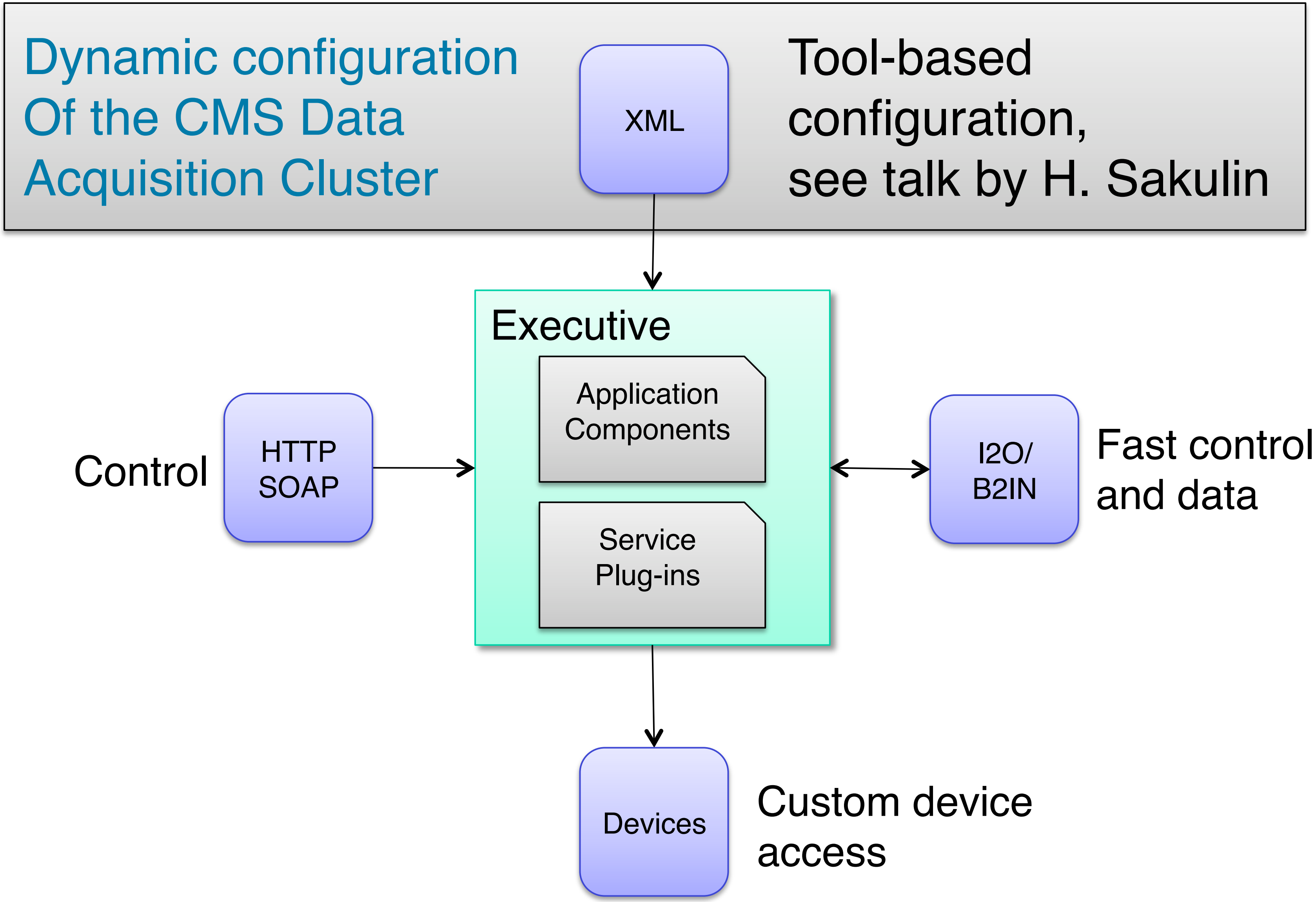
Architecture Foundation

Uniform building block

One or more executives per computer contain application and service components



Architecture Foundation



Dynamic configuration Of the CMS Data Acquisition Cluster

XML

Tool-based configuration, see talk by H. Sakulin

Executive

Application Components

Service Plug-ins

Control

HTTP SOAP

I2O/ B2IN

Fast control and data

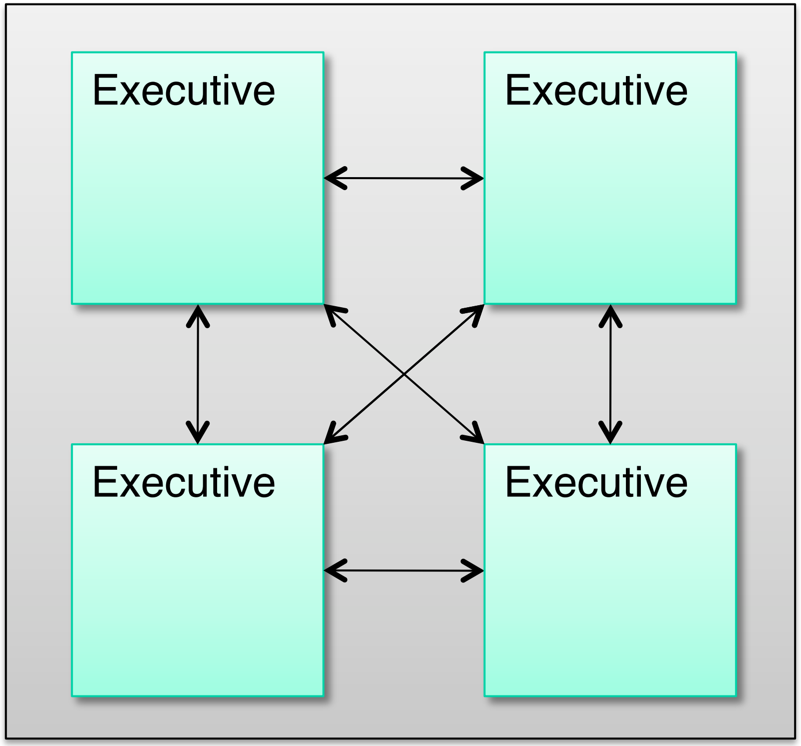
Devices

Custom device access

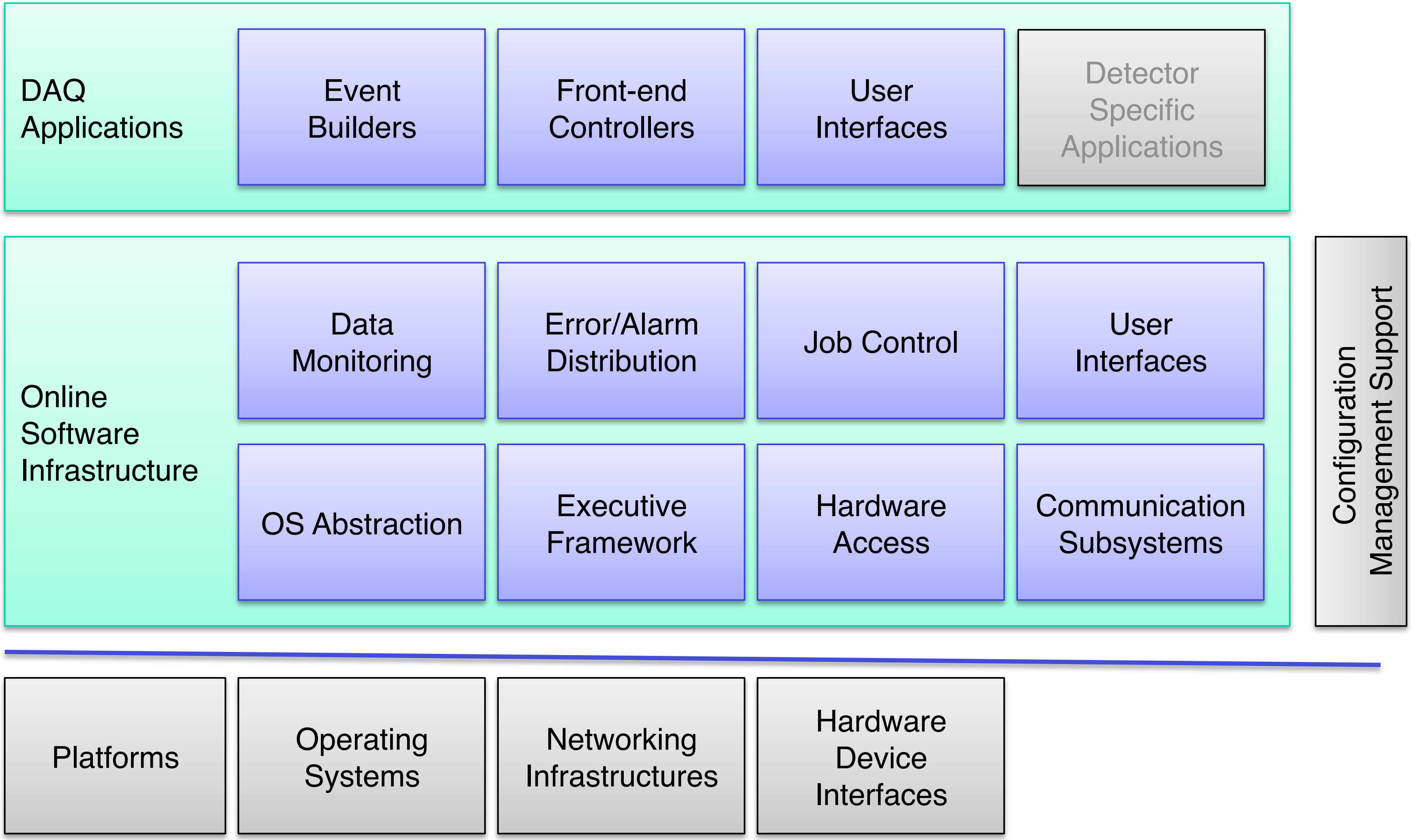
Architecture Foundation

Replicated building blocks

Scalable cluster system architecture



Layered View



DAQ Applications

Event Builders

Front-end Controllers

User Interfaces

Detector Specific Applications

Online Software Infrastructure

Data Monitoring

Error/Alarm Distribution

Job Control

User Interfaces

OS Abstraction

Executive Framework

Hardware Access

Communication Subsystems

Configuration Management Support

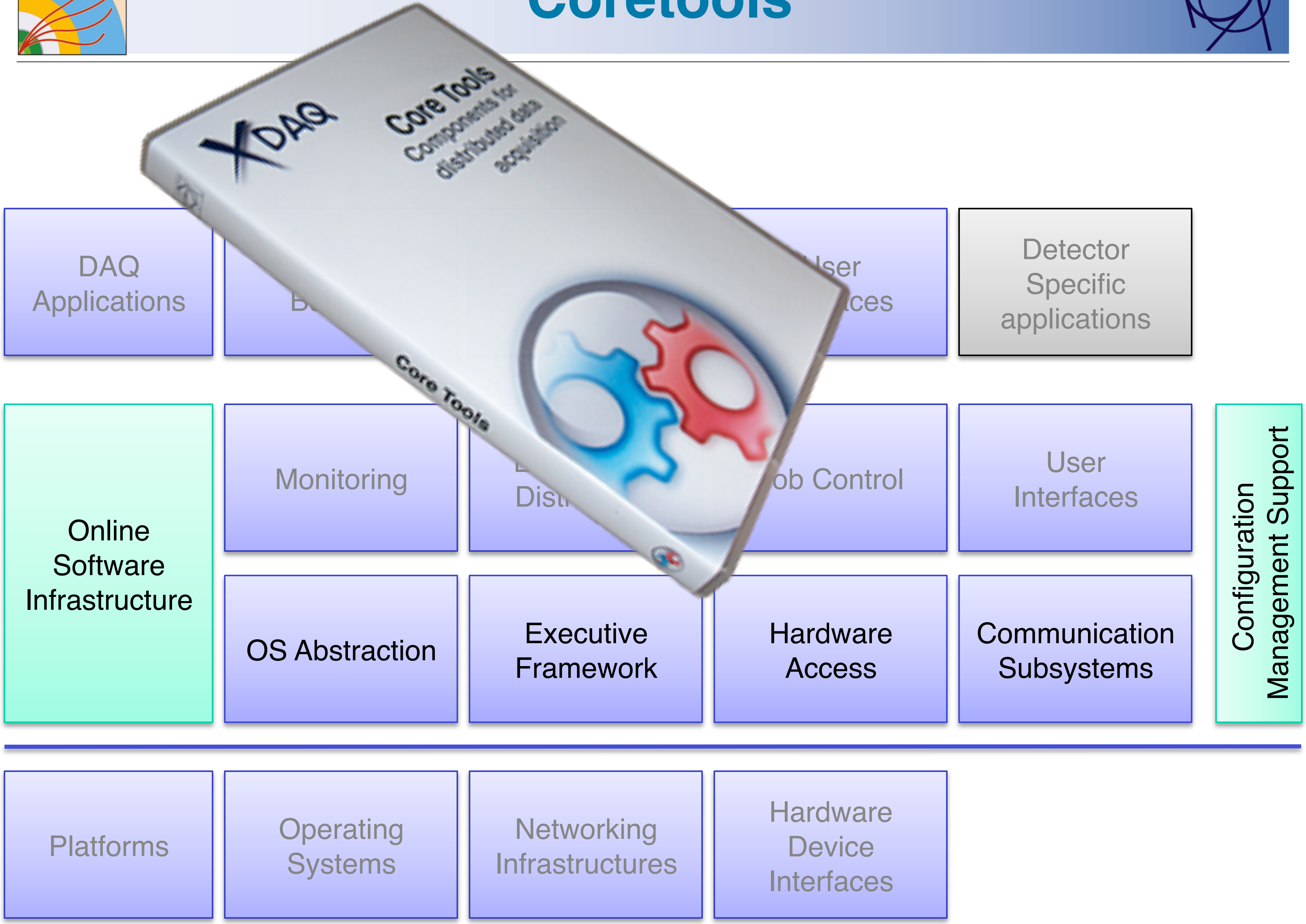
Platforms

Operating Systems

Networking Infrastructures

Hardware Device Interfaces

Coretools



DAQ Applications

B

User
ces

Detector Specific applications

Online Software Infrastructure

Monitoring

D
Dist

ob Control

User Interfaces

OS Abstraction

Executive Framework

Hardware Access

Communication Subsystems

Configuration Management Support

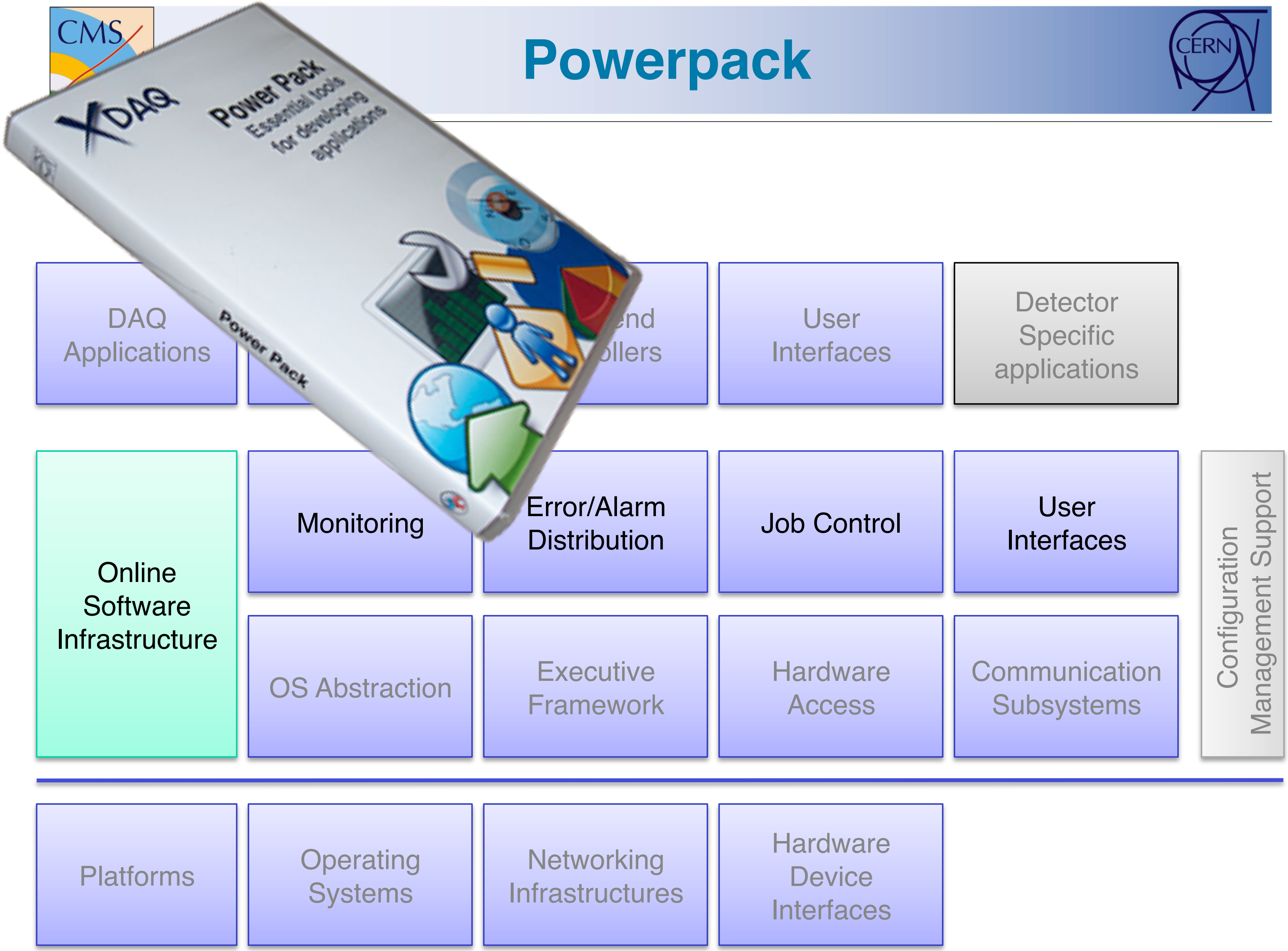
Platforms

Operating Systems

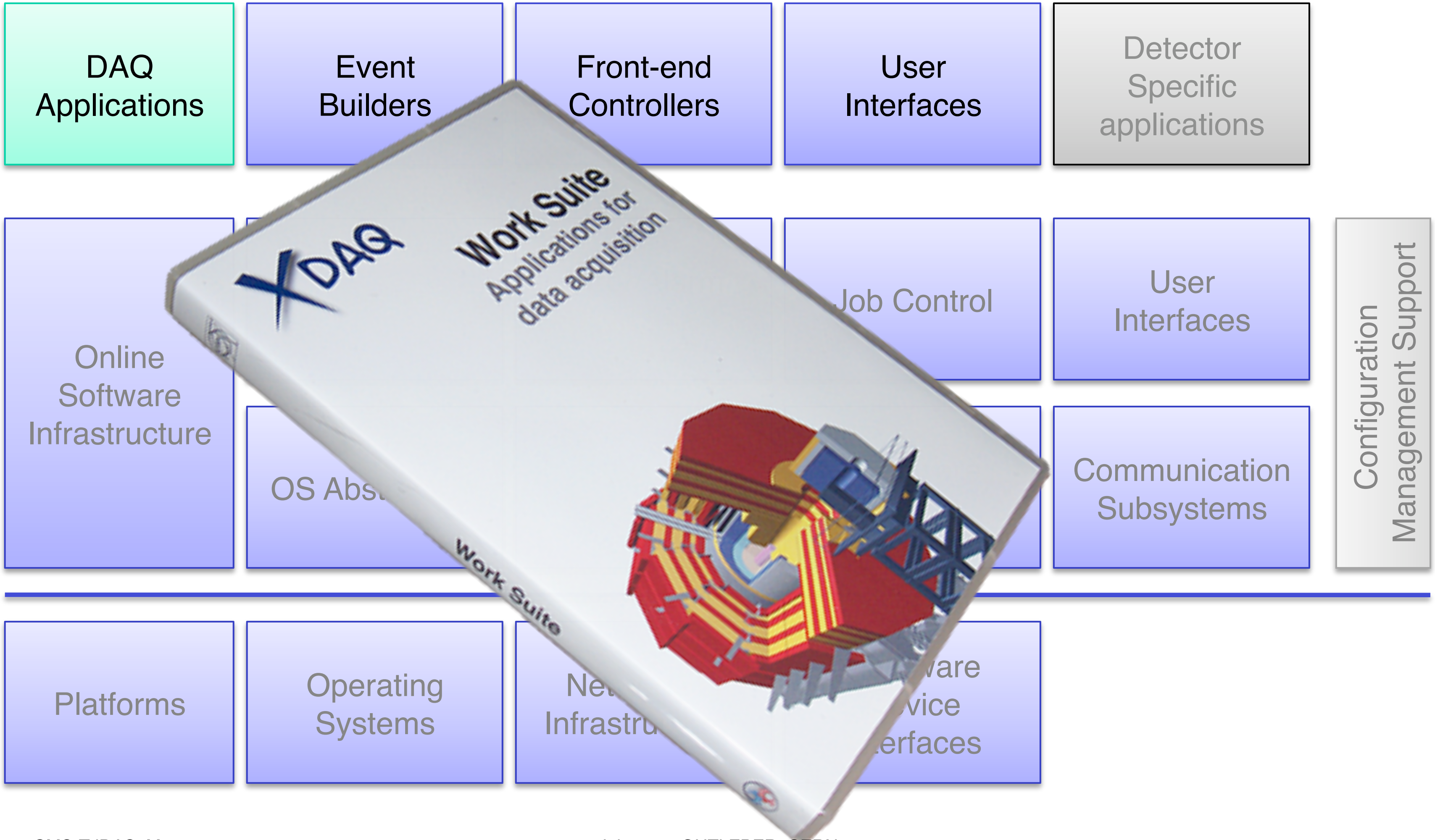
Networking Infrastructures

Hardware Device Interfaces

Powerpack


XDAQ
Power Pack
Essential tools for developing applications
Power Pack
DAQ Applications
nd
ollers
User Interfaces
Detector Specific applications
Online Software Infrastructure
Monitoring
Error/Alarm Distribution
Job Control
User Interfaces
OS Abstraction
Executive Framework
Hardware Access
Communication Subsystems
Configuration Management Support
Platforms
Operating Systems
Networking Infrastructures
Hardware Device Interfaces

Work Suite



DAQ Applications

Event Builders

Front-end Controllers

User Interfaces

Detector Specific applications

Online Software Infrastructure

Job Control

User Interfaces

OS Abs

Communication Subsystems

Configuration Management Support

Platforms

Operating Systems

Ne Infrastru

ware vice erfaces

Codebase

Lines of C++ code

Package	.cc	.h	Sum
Coretools	62’304	43’169	**10’5473**
Powerpack	57’031	18’786	**75’817**
Worksuite	138’166	76’875	**215’041**
Total	**257’501**	**136’636**	**396’331**

Packages

Package	Packages	RPMs
Coretools	9	20
Powerpack	5	34
Worksuite	34	43
Total	**48**	**97**

Personnel

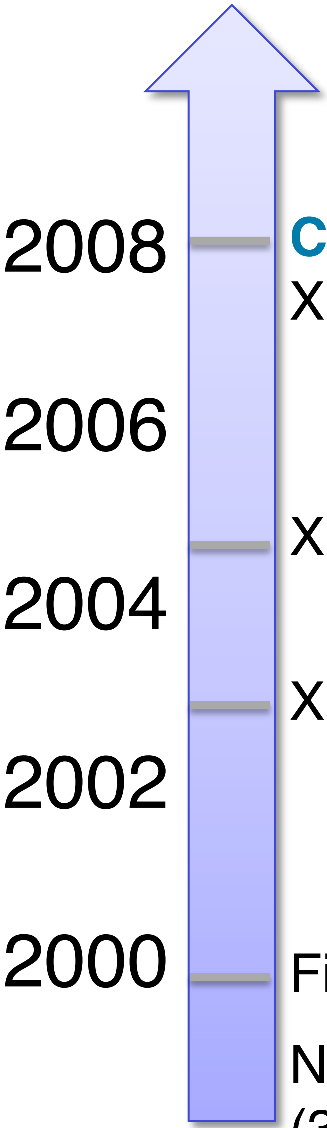
People over the whole development period

Package	FTE	Persons
Coretools/Powerpack	3	≥ 4
Worksuite	5	≥ 6
Total	**8**	≥ **10**

- Figures are given as an indication and do not represent the actual number of employed personnel
- Tasks of personnel includes design, planning, development, documentation, support and consultancy to users
- FTEs are distributed to a changing number of people. About half of the persons were directly employed, the other half was changing personnel
- Production quality code done by long-term employed personnel

Timeline

Well consolidated after eight years
of development and use



Commissioning and first beam event successfully achieved
XMAS – monitoring, orthogonal to applications

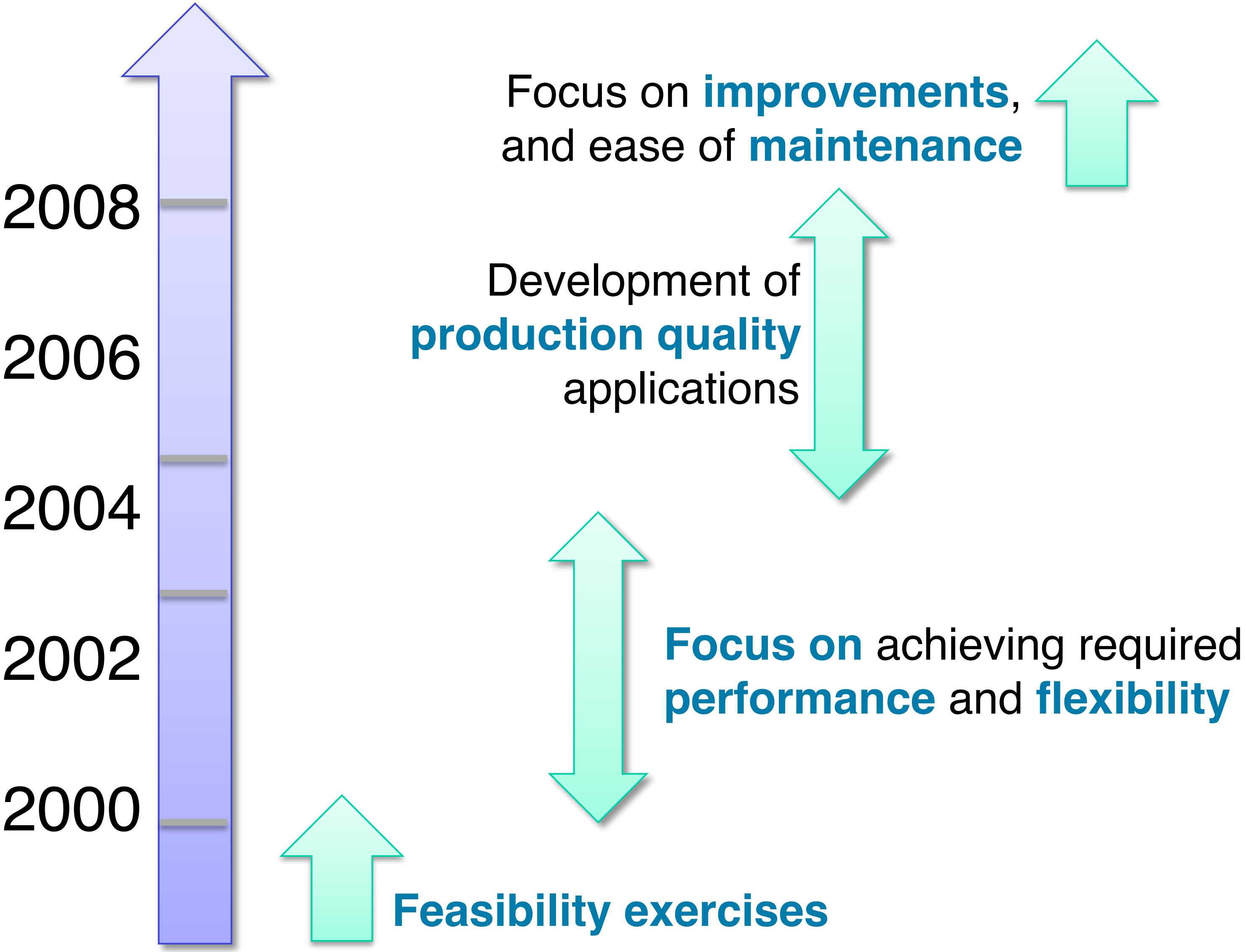
XDAQ 3 – **experiment wide adoption**

XDAQ 2 – Web enters DAQ (SOAP)

First version of XDAQ - I_2O communication kernel

Need for common platform arises
(3 operating systems, many different develoments)

Timeline



Focus on **improvements**, and ease of **maintenance**

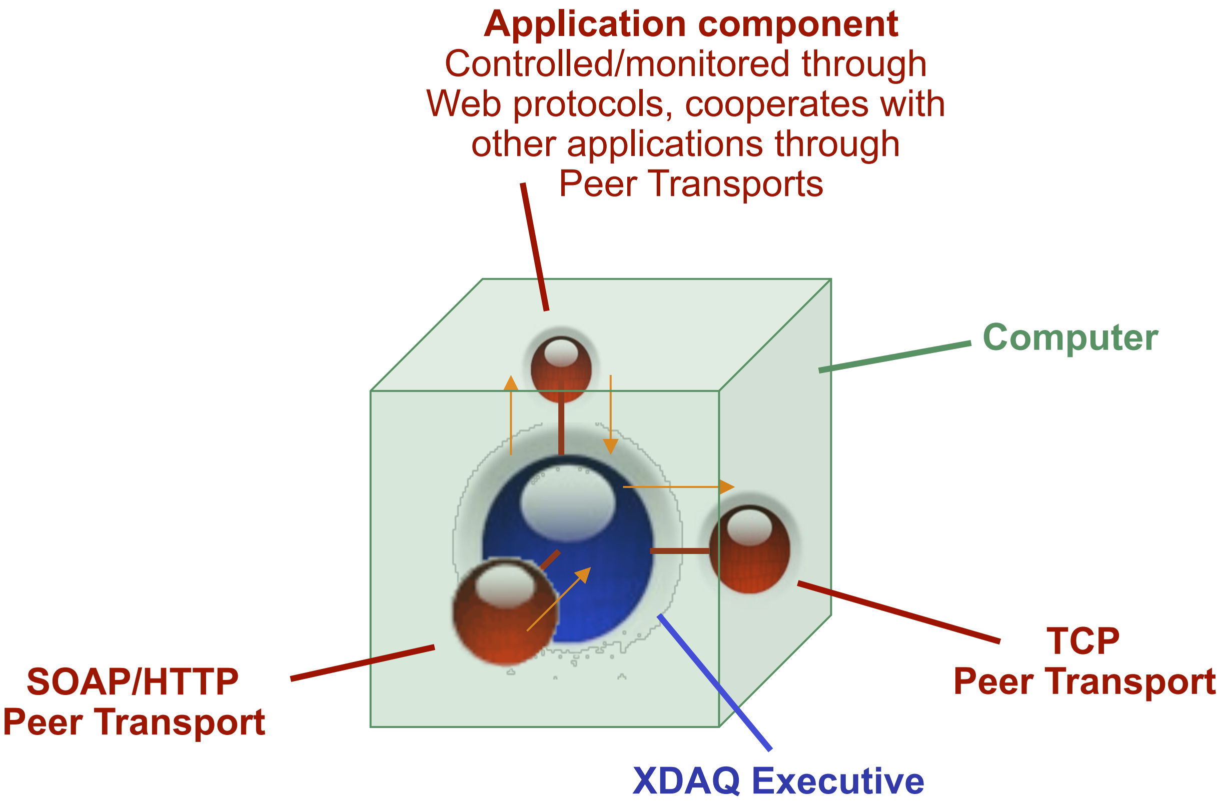
Development of **production quality** applications

Focus on achieving required **performance** and **flexibility**

Feasibility exercises

- Generic event builder
- Web technologies in DAQ

Web Peer (HyperDAQ)


Application component
Controlled/monitored through
Web protocols, cooperates with
other applications through
Peer Transports
Computer
TCP
Peer Transport
SOAP/HTTP
Peer Transport
XDAQ Executive

SOAP/HTTP PT:

Each application becomes browsable through an embedded HTTP/SOAP peer transport.

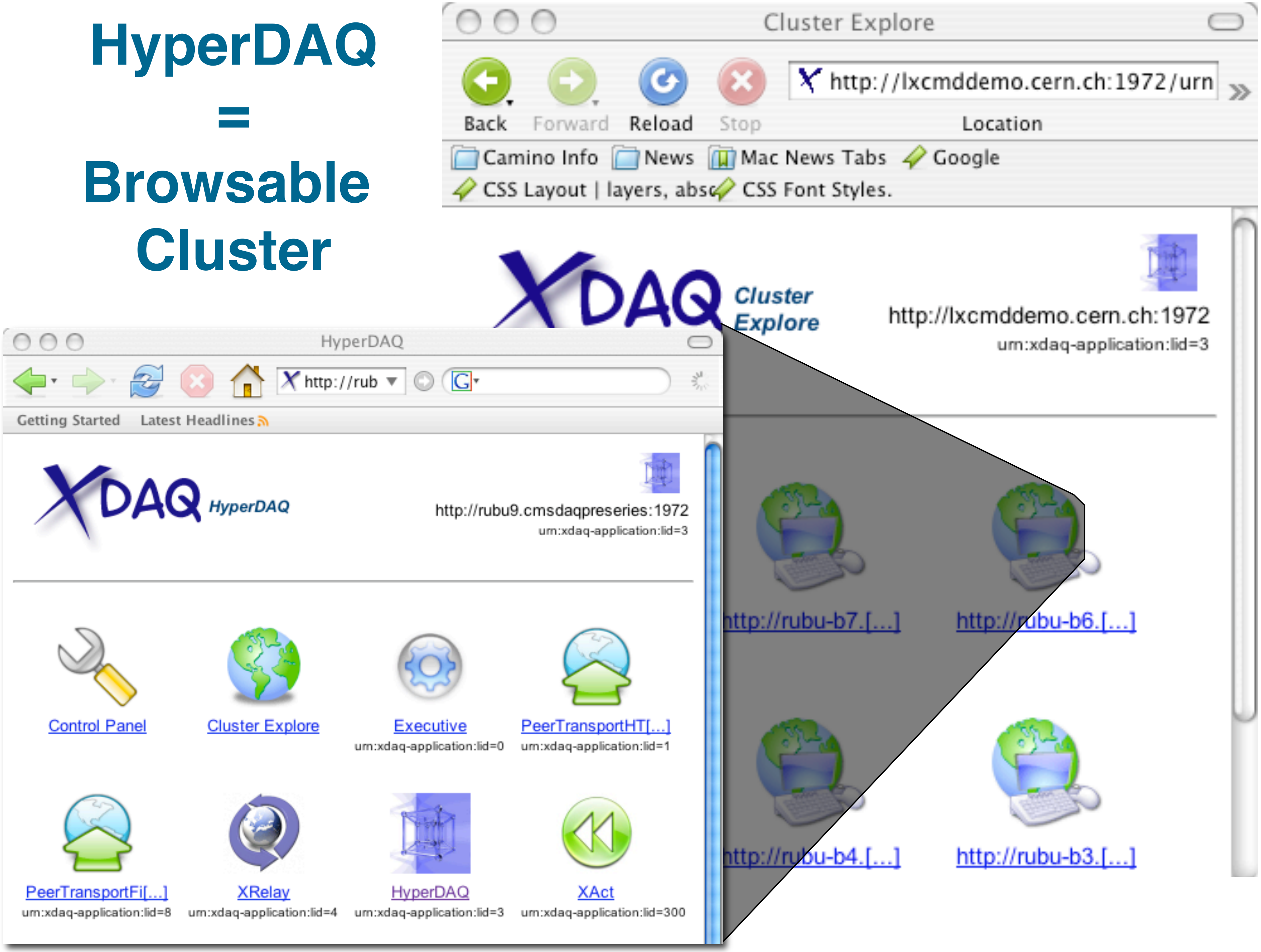
-Navigate directly to each application in the cluster

-Applications implement application specific web pages and callbacks

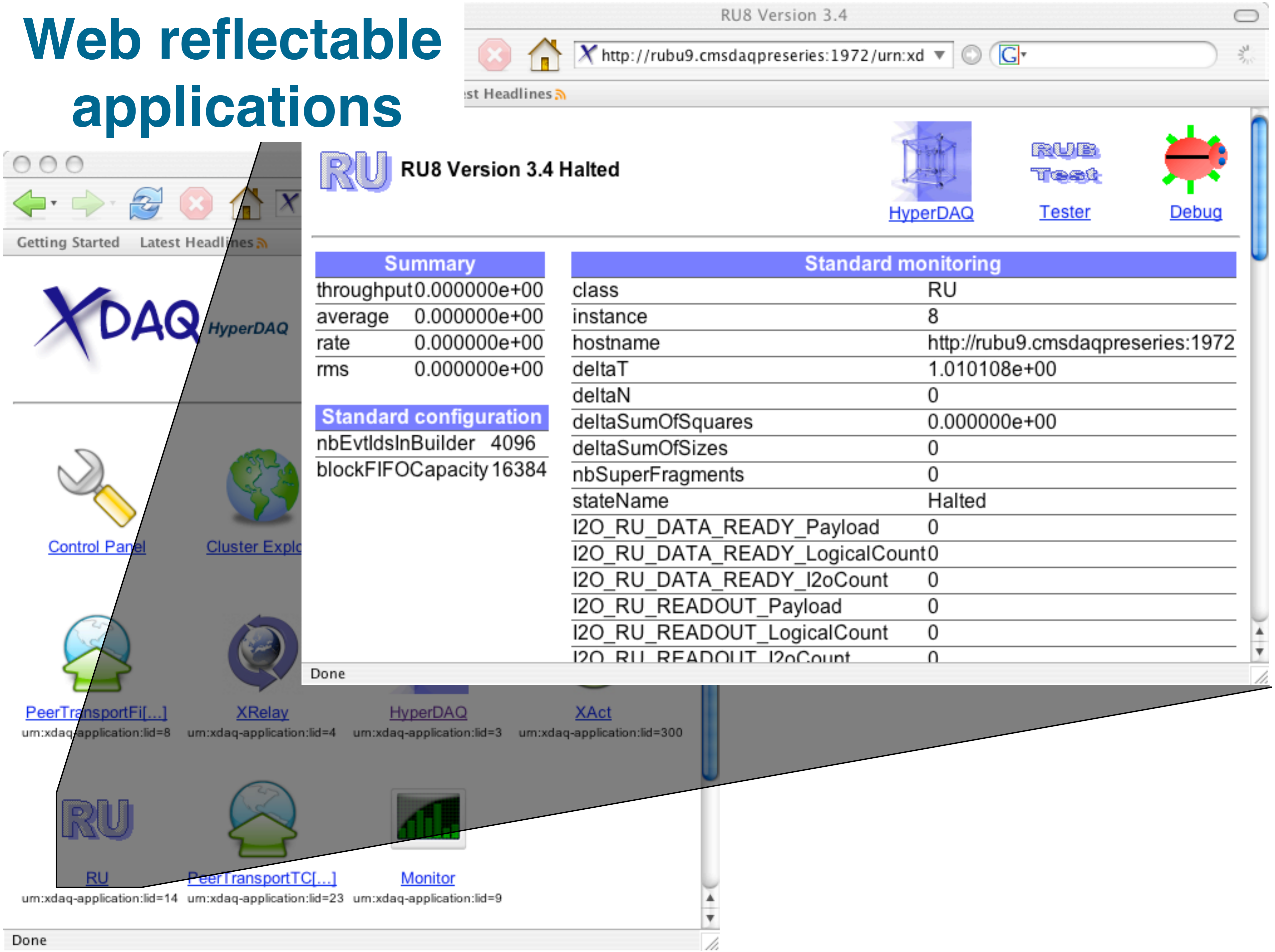
-Components become truly reflective (make themselves visible)

-Gradual integration from manual to automatic through scripts and programmatic approaches

HyperDAQ
=
Browsable Cluster



Web reflectable applications



RU8 Version 3.4
http://rubu9.cmsdaqpreseries:1972/urn:xd
st Headlines
Getting Started
Latest Headlines
XDAQ
HyperDAQ
Control Panel
Cluster Explo
PeerTransportFi[...]
urn:xdaq-application:lid=8
XRelay
urn:xdaq-application:lid=4
HyperDAQ
urn:xdaq-application:lid=3
XAct
urn:xdaq-application:lid=300
RU
urn:xdaq-application:lid=14
PeerTransportTC[...]
urn:xdaq-application:lid=23
Monitor
urn:xdaq-application:lid=9
Done
HyperDAQ
RUB Test
Tester
Debug
Done

RU8 Version 3.4 Halted

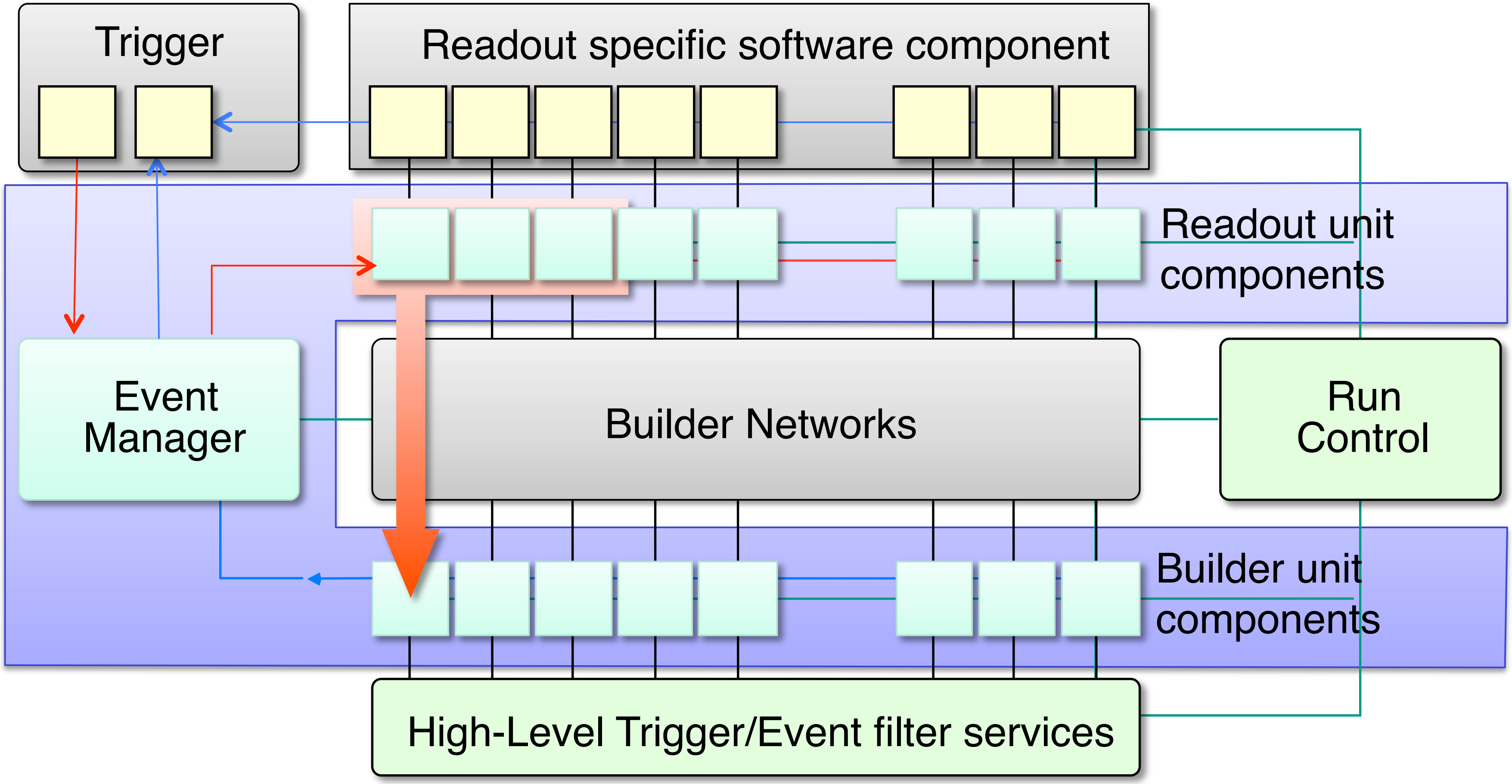
Summary	
throughput	0.000000e+00
average	0.000000e+00
rate	0.000000e+00
rms	0.000000e+00

Standard configuration	
nbEvtIdsInBuilder	4096
blockFIFOCapacity	16384

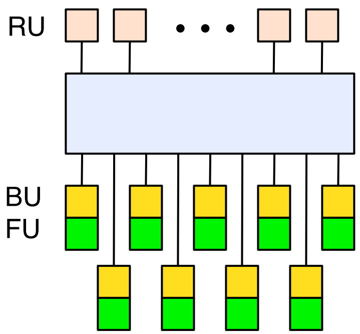
Standard monitoring	
class	RU
instance	8
hostname	http://rubu9.cmsdaqpreseries:1972
deltaT	1.010108e+00
deltaN	0
deltaSumOfSquares	0.000000e+00
deltaSumOfSizes	0
nbSuperFragments	0
stateName	Halted
I2O_RU_DATA_READY_Payload	0
I2O_RU_DATA_READY_LogicalCount	0
I2O_RU_DATA_READY_I2oCount	0
I2O_RU_READOUT_Payload	0
I2O_RU_READOUT_LogicalCount	0
I2O_RU_READOUT_I2oCount	0

Generic Event Builder

- Configurable in size and network technology
- Customization at boundaries through pluggable components



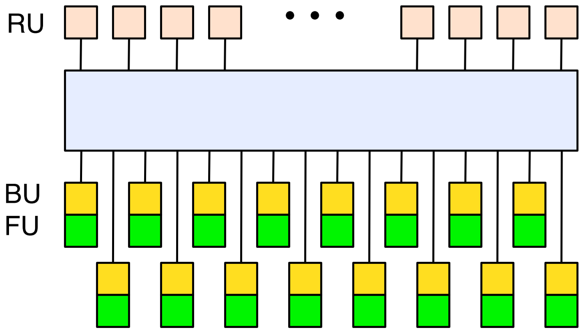
Scaling by Configuration



Reconfigure

System size

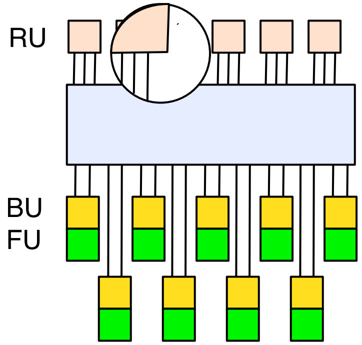
500 RU, 1000s of BU

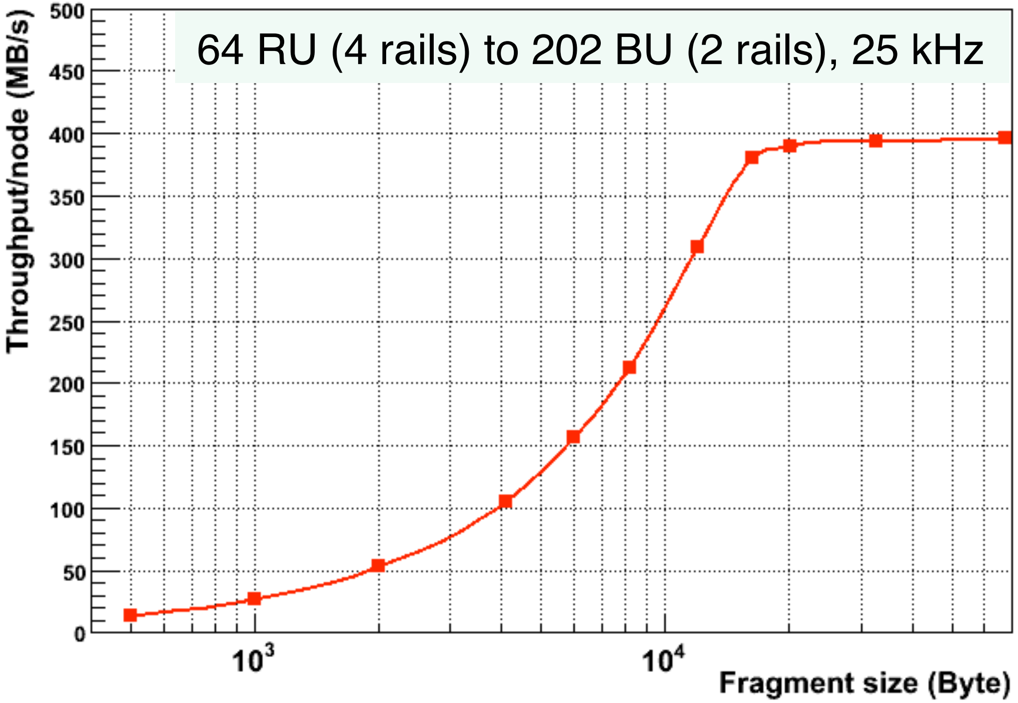


Reconfigure

Communication

1 to 4 GBE ports





Usage Outside CMS

- **PRISMA** and **GASP**
  - DAQ systems have been put in place by INFN Legnaro (Italy) using versions of the CMS online software
- **AGATA**
  - In development
  - New HEP experiment with readout part taking advantage of XDAQ
- **DABC** (GSI)
  - See talk from Hans Essel
  - Data Acquisition Backbone Core
  - Much work on Infiniband done

Summary

- Concentrate DAQ **expert knowledge in one place**
  - Software used in larger context is better understood
  - Avoid repeating same mistakes
- Provide **uniform infrastructure** to all groups in the experiment that need DAQ functionality
  - Work by product not by specification
  - Stability and fitting the requirements are key
- Ensure that **good efficiency** is achieved **under different operating conditions**

Conclusion

Strive towards building a DAQ system as a process of **assembly re-usable components** in a predetermined way **rather than** a **programming** task.

Achieved **a uniform DAQ product-line** for all CMS data acquisition application scenarios ranging from single CPU setups to the final systems comprising thousands of nodes.

http://cern.ch/xdaq