Summary of DAQ Session

David Cussans, Paris, Palaiseau, October 2007

Outline

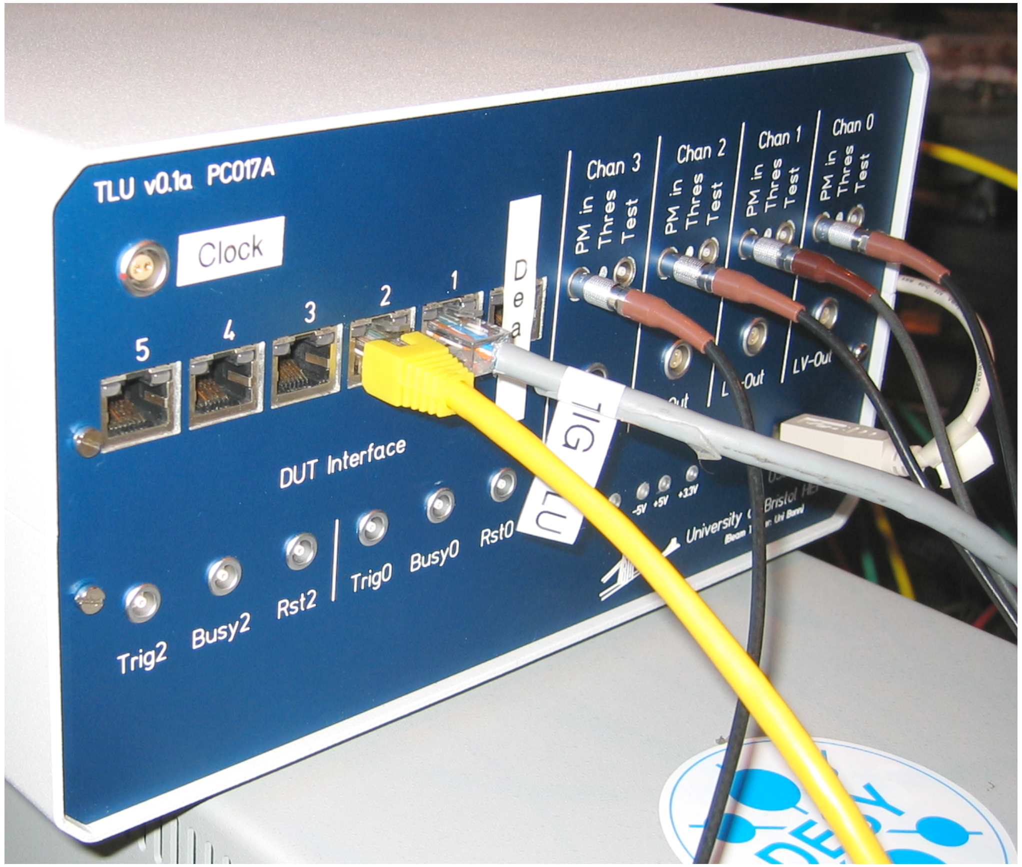
- Summarize "Common DAQ Session"
  - Beam-test trigger.
  - Beam-telescope DAQ
  - TPC DAQ
  - Calice DAQ
    - Overview of current status
    - Review of possible frameworks
  - Conclusions.

Trigger Logic Unit

- Needed simple trigger for beam-tests.
  - Spent some work to make a more general “Trigger/Timing Logic Unit” (TLU)
- Accepts triggers from e.g. beam-scintillators
- Drives trigger/busy logic to detectors
- Stores time-stamp (~ 2.5ns resolution)

Trigger Logic Unit

- Five “prototypes” built
- Cost ~ €1k each

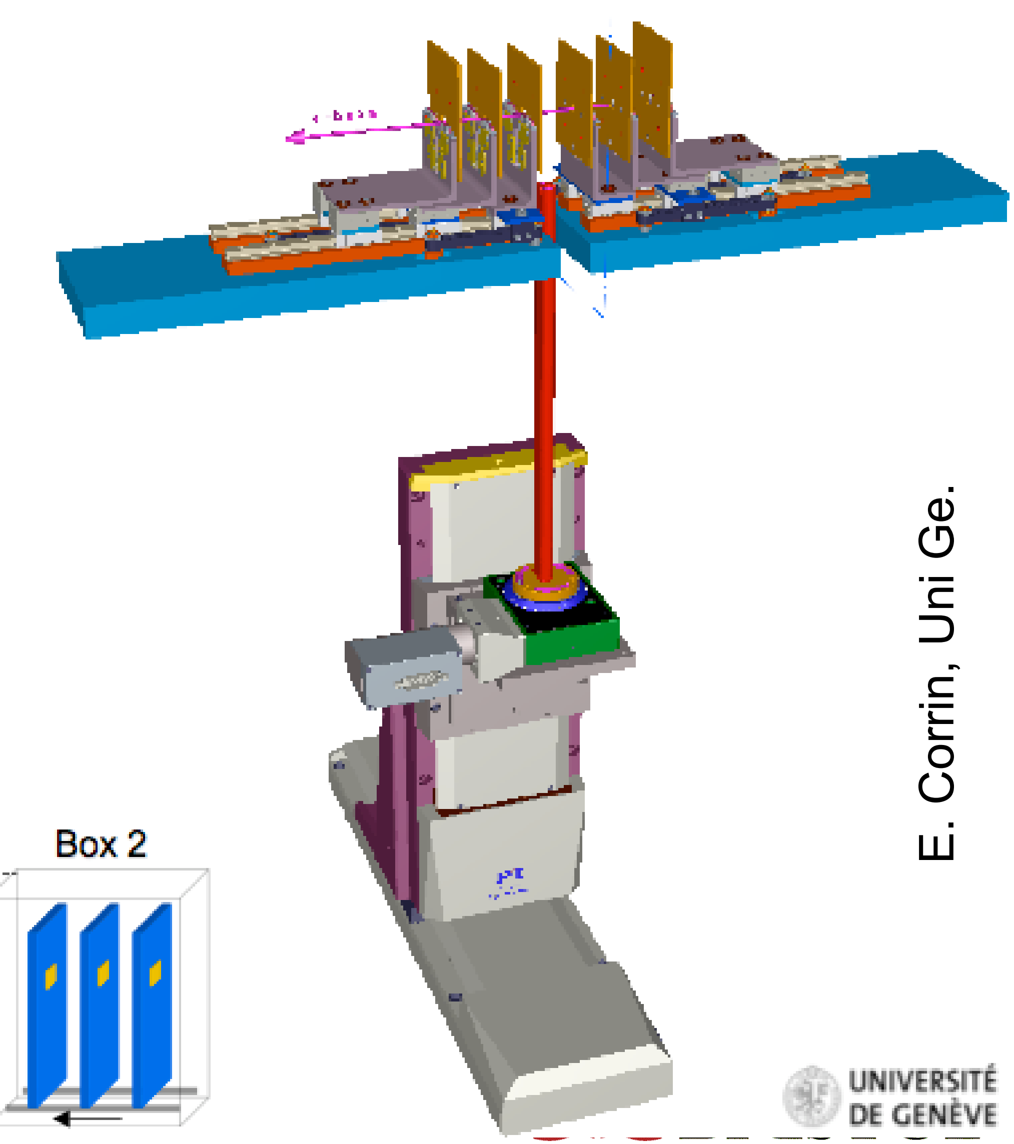


Trigger Logic Unit

- TLU exists and has been used in JRA1 beam-tests
- TLU being investigated by JRA2, interest from JRA3.
  - Firmware modified for fixed latency, jitter 25ps
- Aim to make simple enough to use so won't even need a NIM crate.

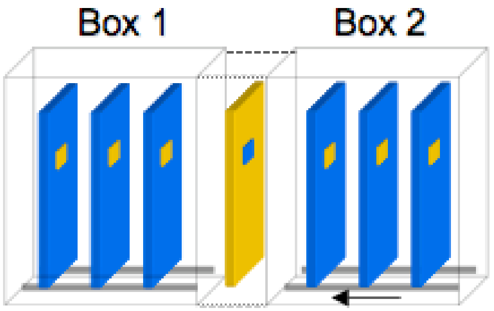
The JRA1 Beam Telescope

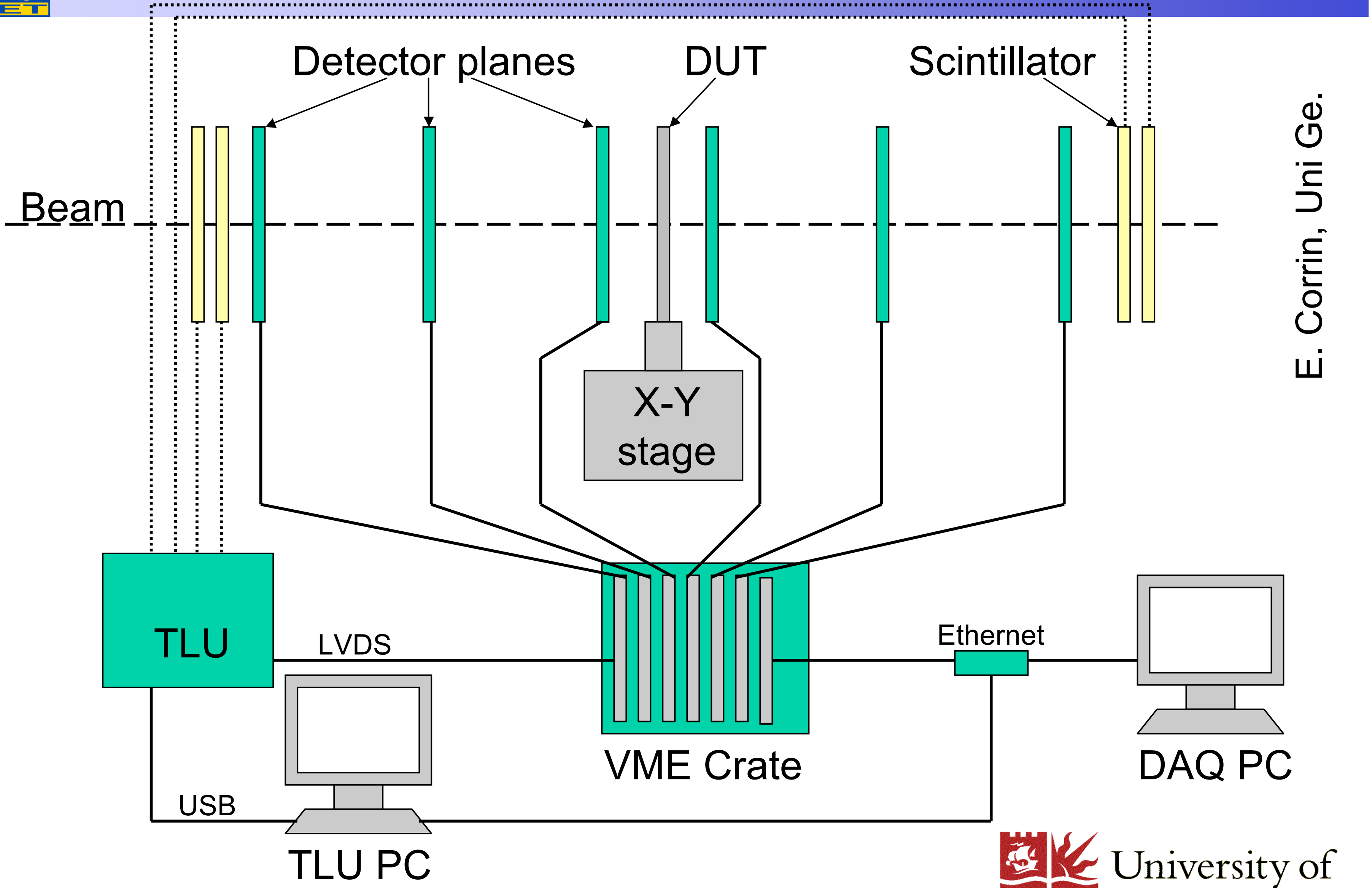
- 6 telescope planes
- DUT movable
  via X-Y table
- Cooling can be provided
- Flexible geometry
- High-resolution planes
  close to DUT possible



E. Corrin, Uni Ge.

UNIVERSITÉ DE GENÈVE





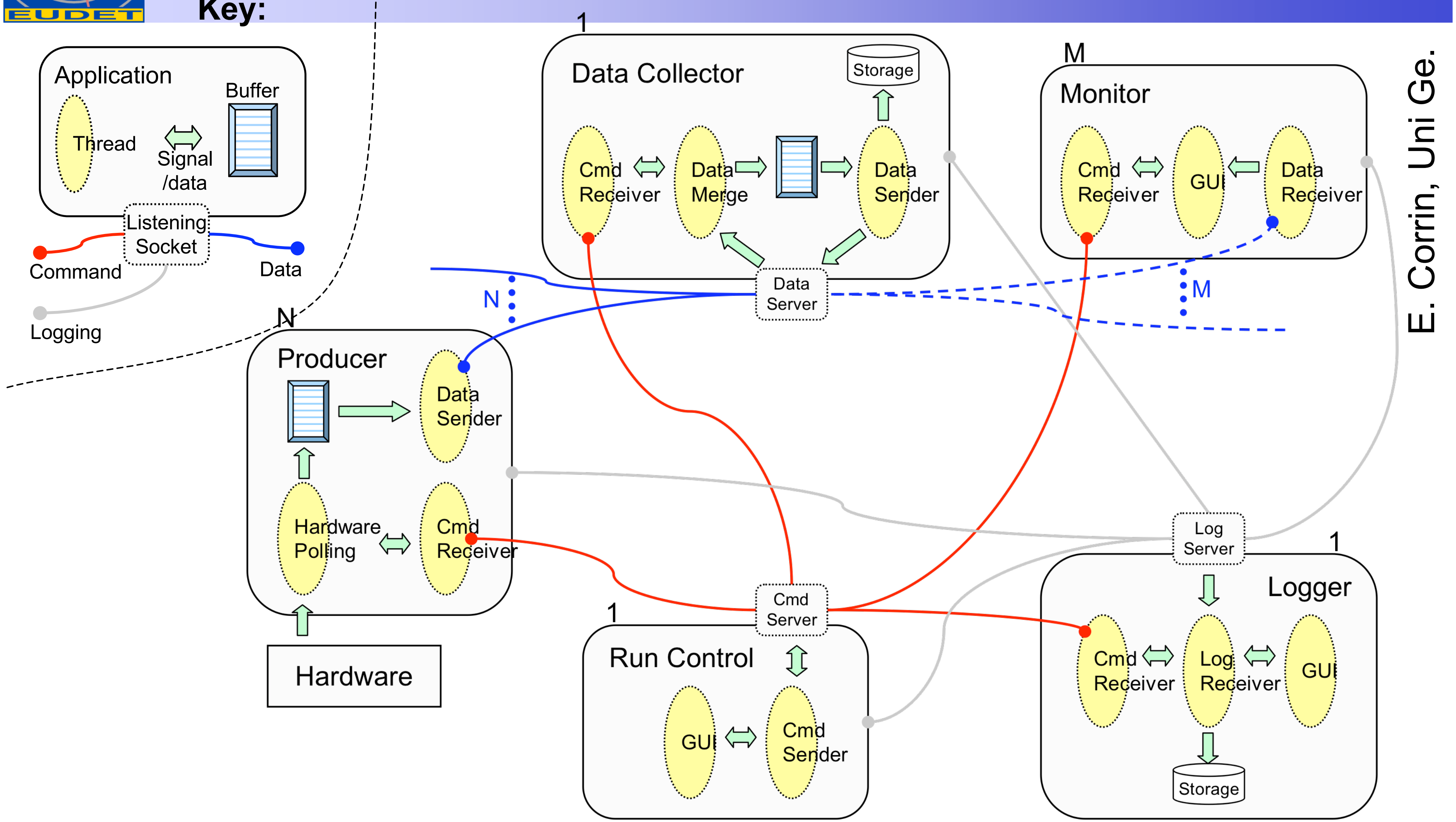
E. Corrin, Uni Ge.

JRA1 DAQ Software

- Custom DAQ software written in C++
- Uses POSIX for threads and sockets
- Run Control GUI using Qt
- Online Monitor using Root
- Runs on Mac OS X, Linux, and Windows (using cygwin)
- Highly modular, allowing DUTs to be easily integrated into the DAQ

E. Corrin, Uni Ge.

JRA1 Software Architecture



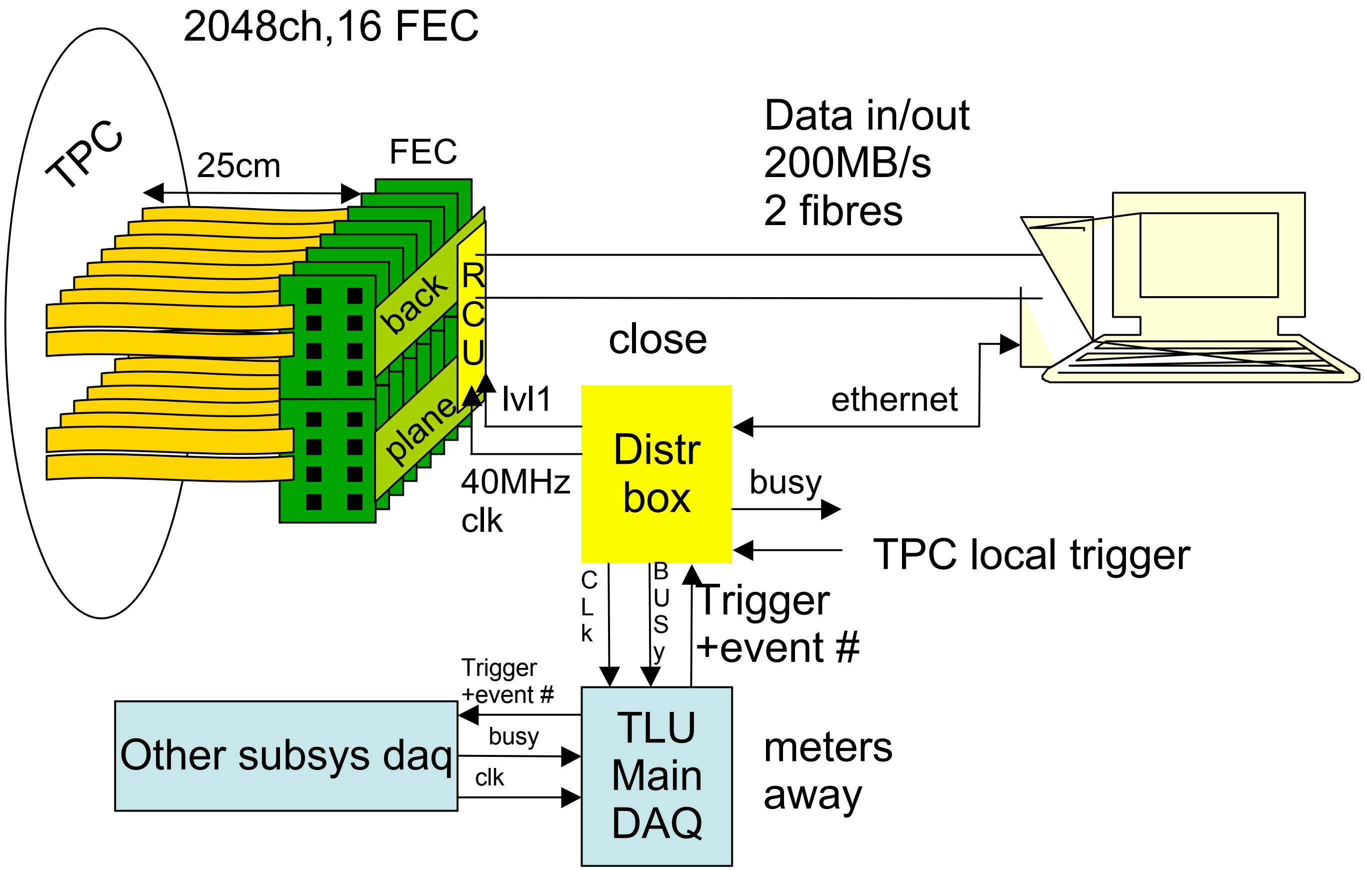
E. Corrin, Uni Ge.

David Cussans, Paris, Palaiseau, October 2007

JRA1 DAQ

- Reasonably stable and usable DAQ system exists now.
- A few minor issues (speed, stability) but they are being worked on
- Easy to integrate other DUTs into the framework
- Documentation being improved.

JRA2: TPC



Ulf Mjörnmark, LUND

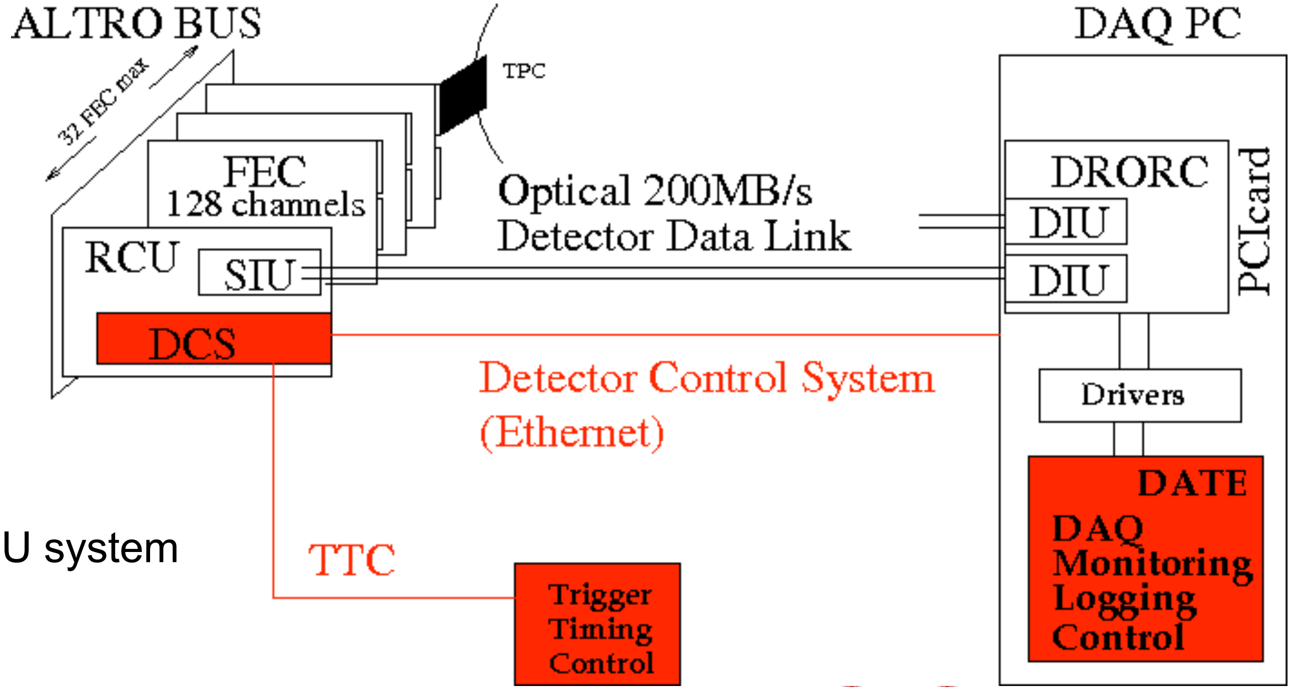
David Cussans, Paris, Palaiseau, October 2007

JRA2 DAQ

Ulf Mjörnmark, LUND

Based on the ALICE TPC readout:

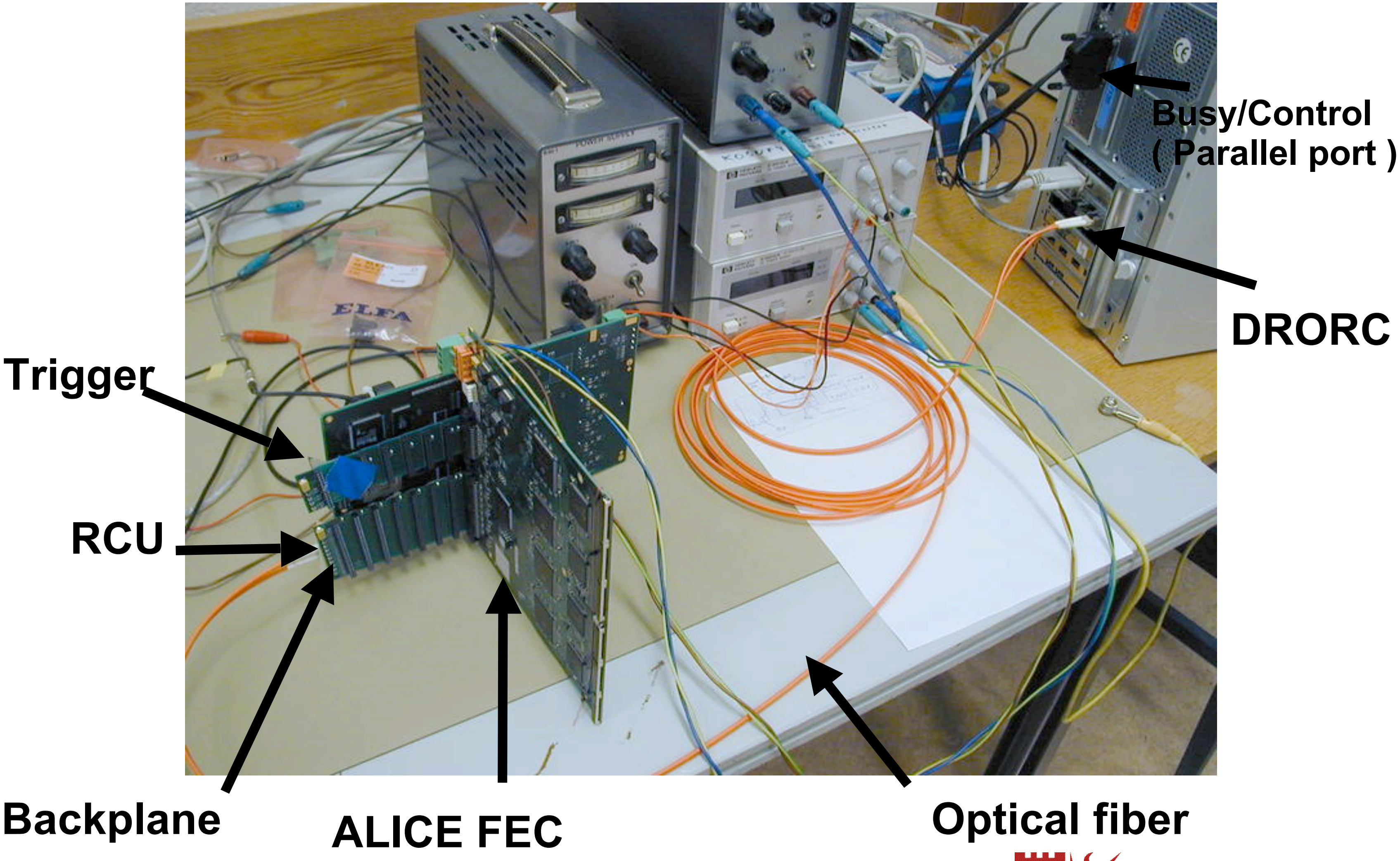
Front End Card (FEC), to be modified for new amplifier
Readout Control Unit (RCU), modified for clock/trigger/25Mhz sample clock
Source Interface Unit (SIU)
Read Out Receiver Card (DRORC), Destination Interface Unit (DIU)
ALICE API/drivers
Build our own DAQ on top
Distributor Box (DBOX) to distribute clock/trigger/busy
Interface to common DAQ



EUDET: 1 RCU
10000 ch: 4 RCU
possible to distribute 1 RCU system

David Cussans, Paris, Palaiseau,

JRA2 DAQ Hardware tests



Busy/Control
(Parallel port)

DRORC

Trigger

RCU

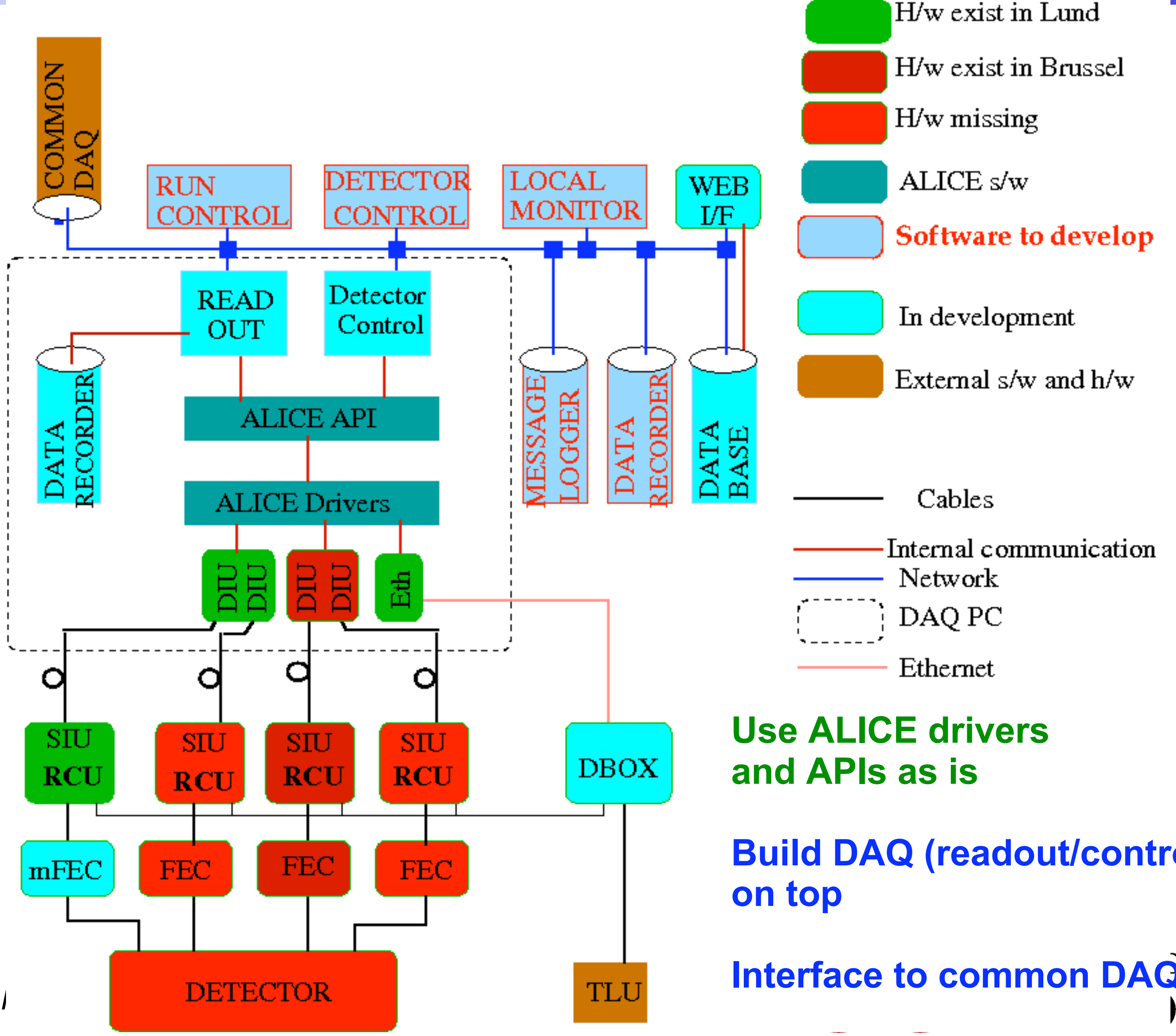
Backplane

ALICE FEC

Optical fiber

JRA2 DAQ S/ware Architecture

Ulf Mjörnmark, LUND



Use ALICE drivers and APIs as is

Build DAQ (readout/control) on top

Interface to common DAQ

JRA2 DAQ Summary

- Based on ALICE TPC readout
- New preamplifier on front end card
- Using ALICE drivers
- Build simple DAQ on top
- Distributor box to distribute trigger/clock/busy

Simple test setup working

Work in progress on:

- front end cards
- distributor box
- readout of hardware
- readout configuration

Missing:

- run control
- monitoring
- detector control
- data transfer and format

Ulf Mjörnmark, LUND

David Cussans, Paris, Palaiseau, October 2007

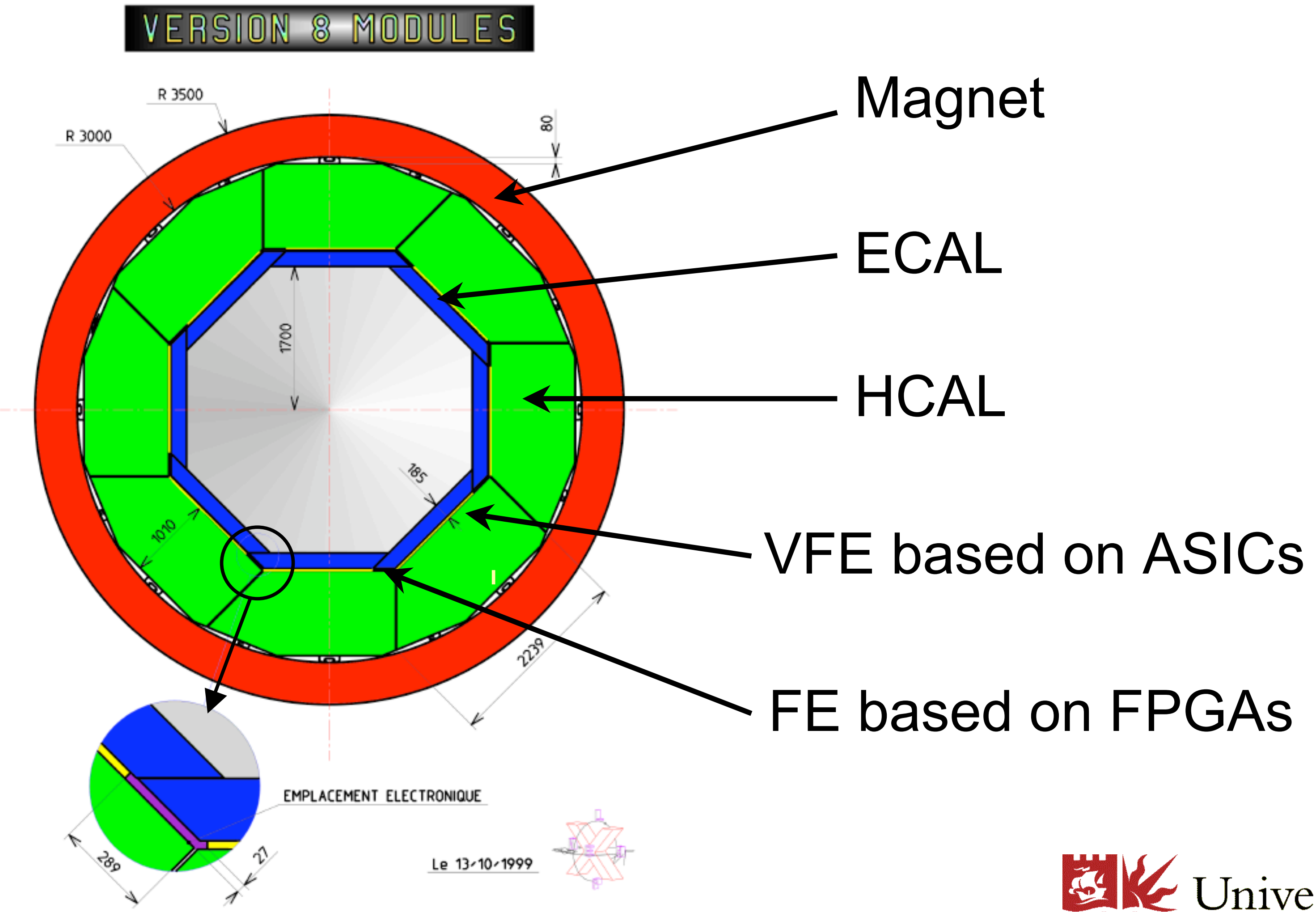
University of BRISTOL

JRA3: Calorimetry

- Use commercial components where possible:
  - Readout links use standard connectors and protocols
  - PCs with PCI (-Express) cards
- Modular/Generic Structure
  - Generic readout system as much as possible
  - Detector specific interfaces only at ends of chain
  - Other ‘bespoke’ functionality in firmware
- Clock and Control attempts commercial hardware too
  - Extract ‘fast’ signals from commercial signalling
- DAQ software generic for all detectors
  - Try use something off-the-shelf …

Matthew Warren, UCL

Calo Detector Layout



Magnet

ECAL

HCAL

VFE based on ASICs

FE based on FPGAs

DAQ architecture

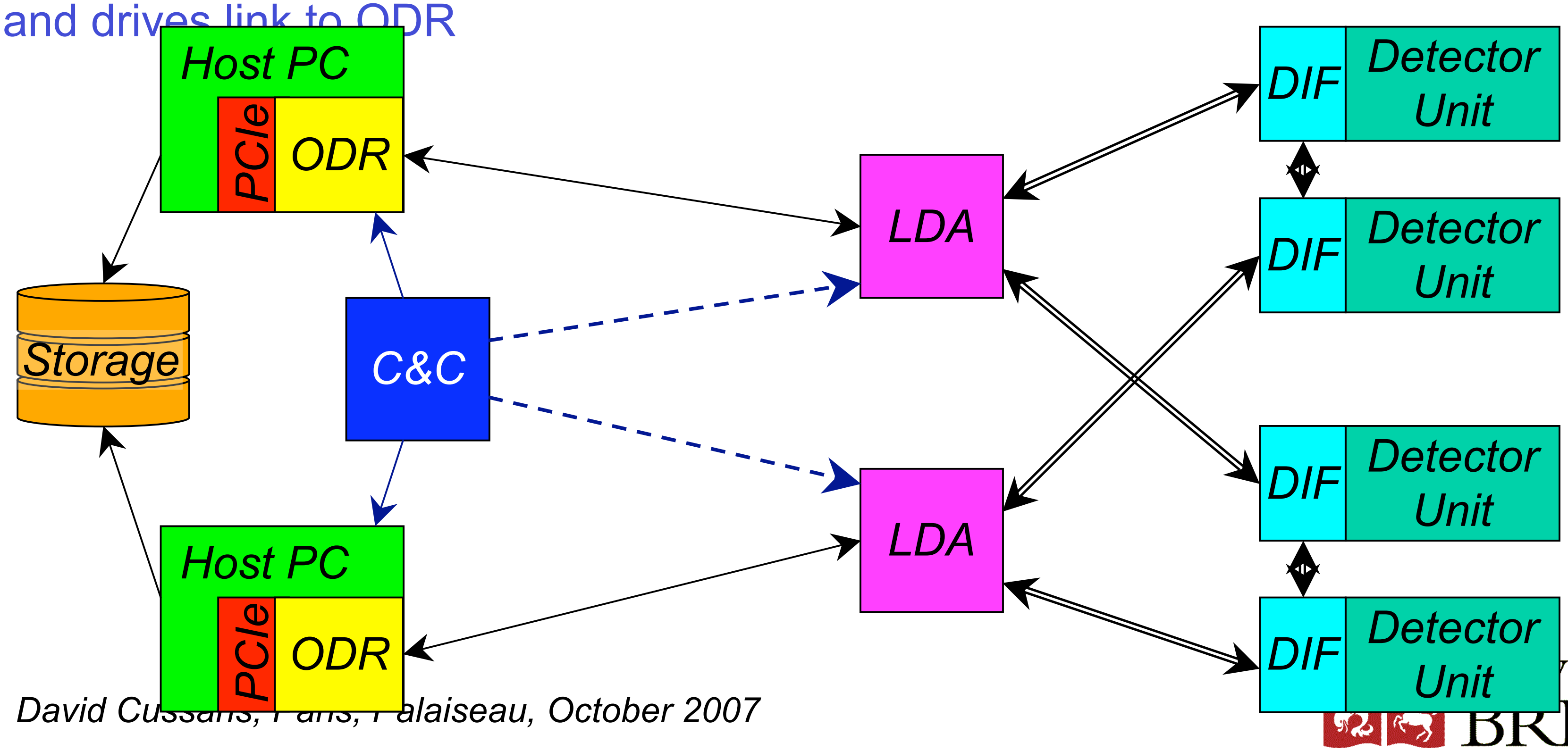
Detector Unit: ASICs

DIF: Detector InterFace connects Generic DAQ and services

LDA: Link/Data Aggregator – fanout/in DIFs and drives link to ODR

ODR: Off Detector Receiver – PC interface for system.

C&C: Clock & Control: Fanout to ODRs (or LDAs)



JRA3 DAQ H/ware Overall Status

ODR: Stage 1 complete – Ethernet based read-out link

- Stage 2 starting – Control up-link + more channels
- Receiver ready now. Bi-directional Q2 2008

LDA: main hardware purchased, DIF/ODR interface hardware add-ons in-progress at manufacturer.

- Interfaces specified
- Working prototype Q2 2008

DIF: Generic DAQ Interfaces mostly defined, plenty of detector specific interfaces

C&C: Hardware design in progress now – Comments??

- Hardware for Q1 2008

Software:

-Review of frameworks underway

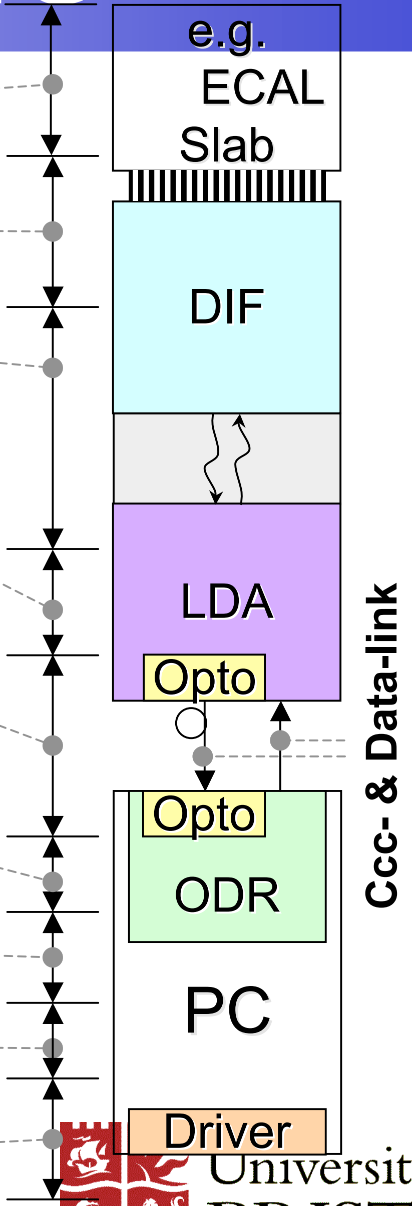
Schedule:

- Development prototypes mid-2008
- FULL DAQ mid-2009

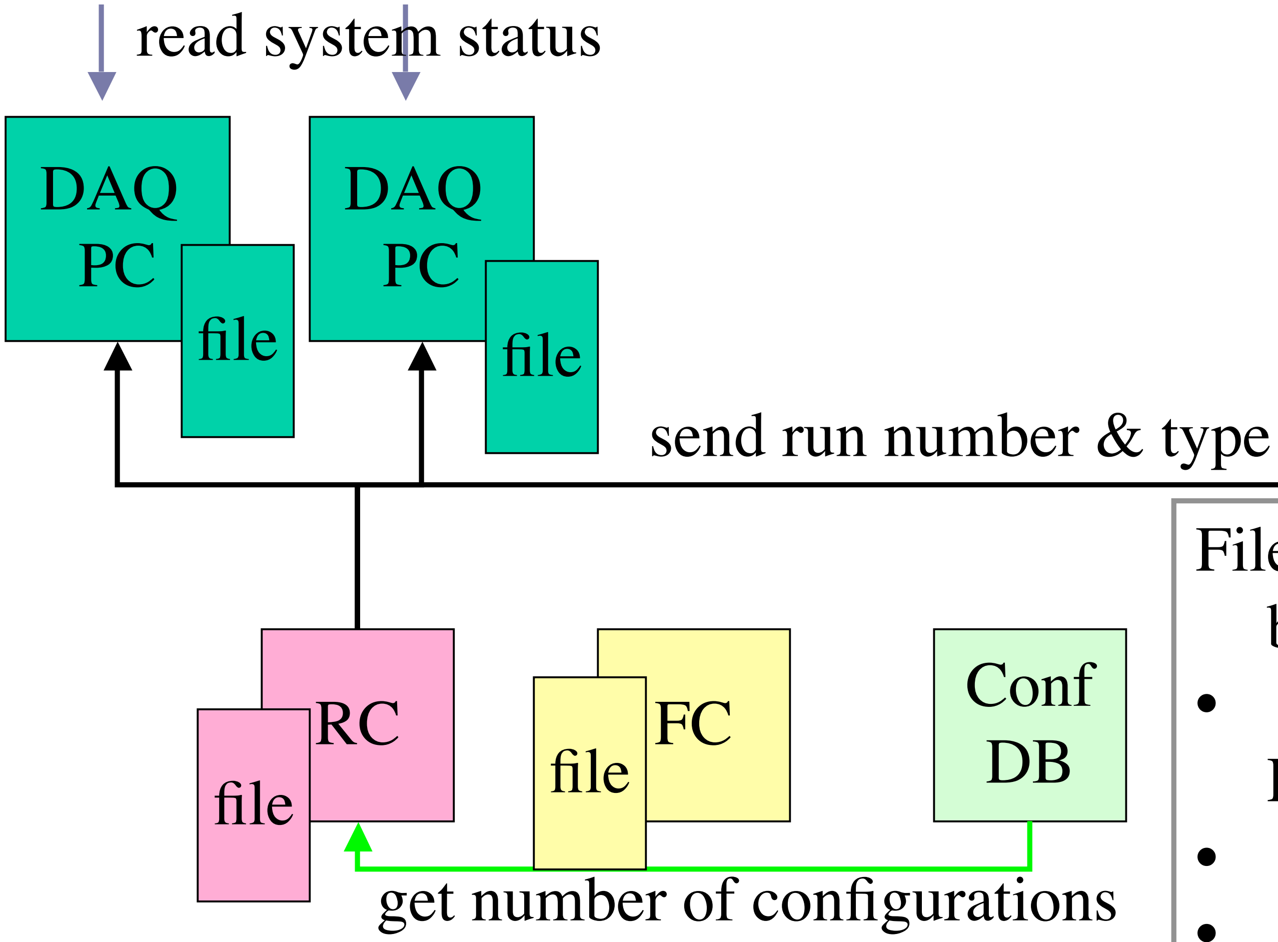
JRA3 DAQ Architecture

Tau Wu, RHUL

- Slab hosts VFE chips
- DIF connected to Slab
- LDA servicing DIFs
- Link/Data Aggregator (LDA)
- LDAs read out by ODR via opto-links
- Off-Detector Receiver (ODR)
- PC hosts ODR
  - PCI-Express driver software
- Local Software DAQ
- Full blown Software DAQ



DAQ software for EUDET



Files to be written for book keeping:

- system status by DAQ PC
- run info by RC PC
- system status by FC

What DAQ software should be used?

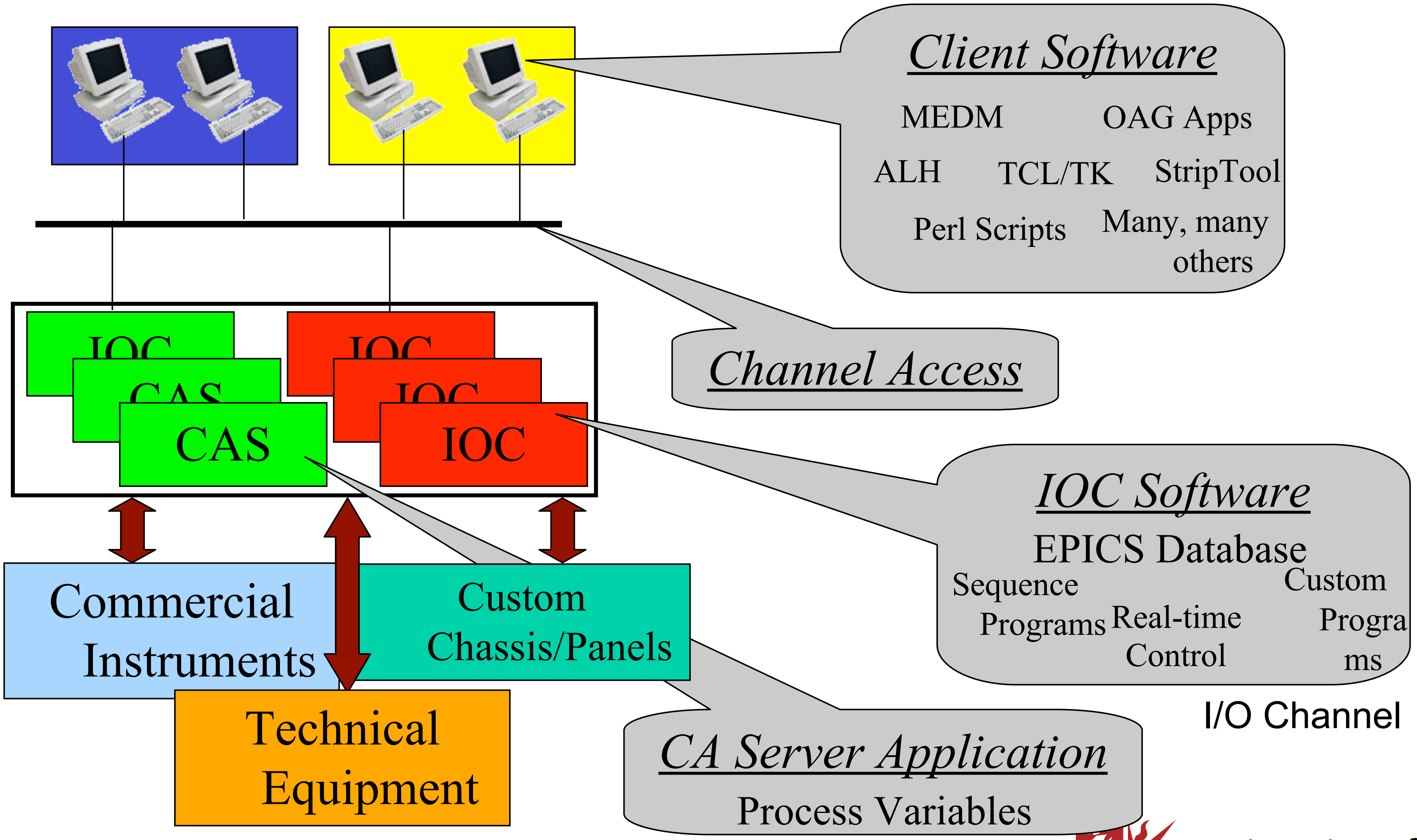
- An effort is focused on EPICS;
- An alternative candidate is ACE;
- Looking into DOOCS framework.

DAQ software candidate: EPICS

- What’s EPICS: **E**xperimental **P**hysics & **I**ndustrial **C**ontrol **S**ystem
- A World-wide Collaboration (LANL, SLAC, JLAB, DESY, KEK, RAL, PSI, APS…)
- A Control System Architecture
  - Network-based “client/server” model with Channel Access Protocol for passing data
  - A distributed real-time database of machine values
- A Software Toolkit: A collection of software tools, comprehensive and scalable control system
- Successful cases: STAR/D0 …

http://www.aps.anl.gov/epics/

Canonical Form of an EPICS Control System



Client Software

MEDM

OAG Apps

ALH

TCL/TK

StripTool

Perl Scripts

Many, many others

Channel Access

IOC

CAS

CAS

IOC

IOC

IOC

IOC Software

EPICS Database

Sequence Programs

Real-time Control

Custom Programs

I/O Channel

Commercial Instruments

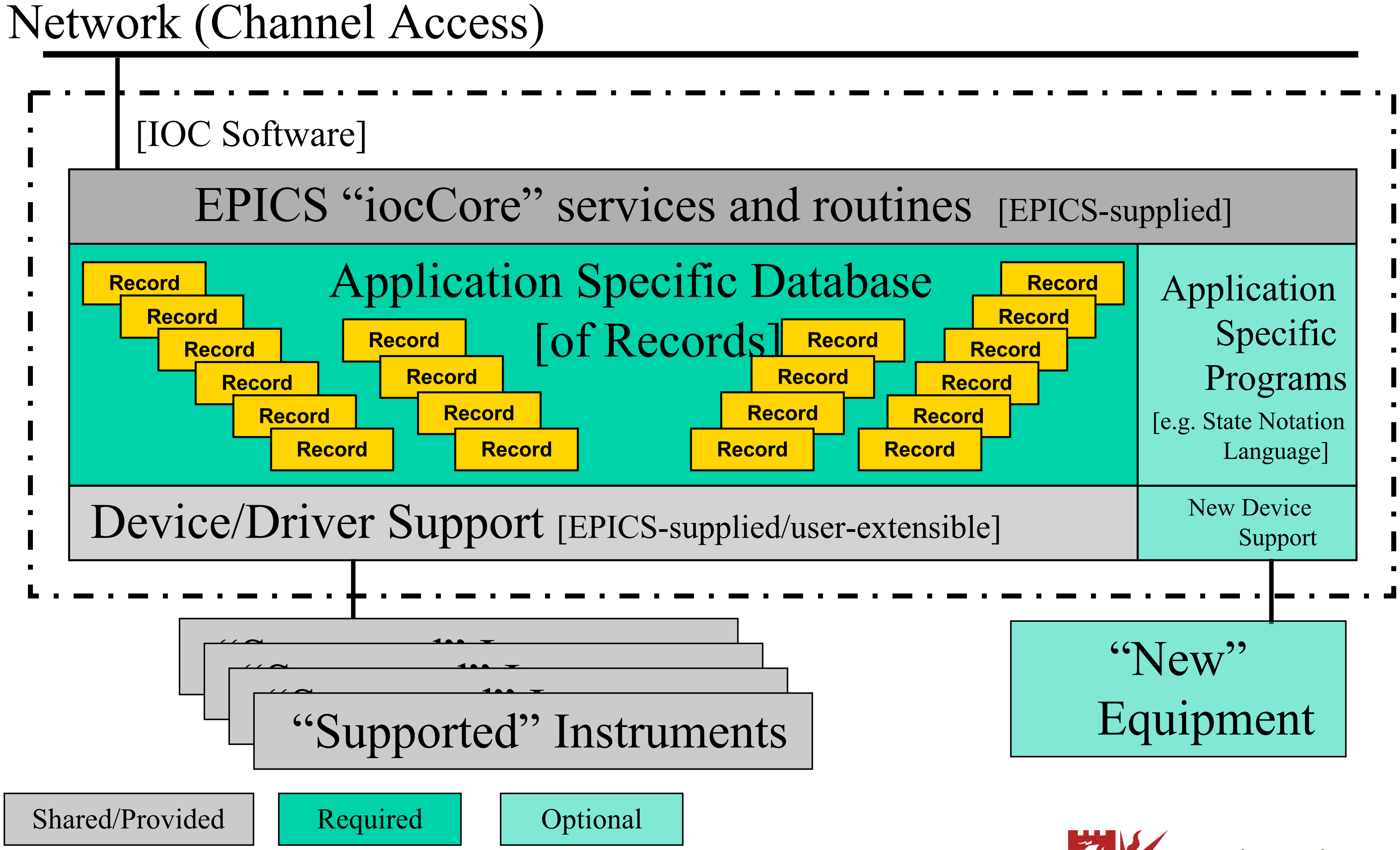
Custom Chassis/Panels

Technical Equipment

CA Server Application

Process Variables

EPICS IOC Software in One Slide

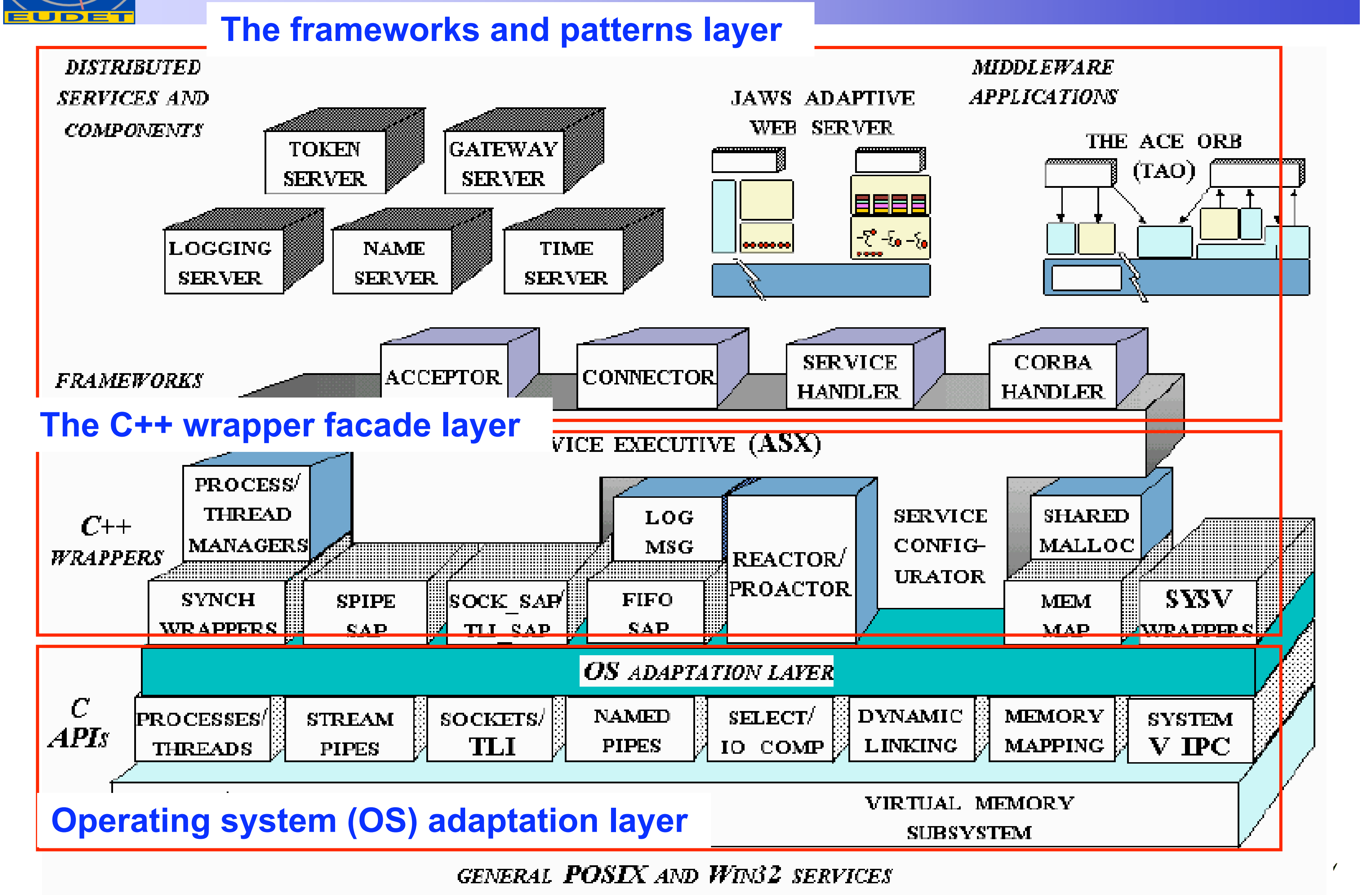

Network (Channel Access)
[IOC Software]
EPICS “iocCore” services and routines [EPICS-supplied]
Application Specific Database [of Records]
Record
Application Specific Programs [e.g. State Notation Language]
Device/Driver Support [EPICS-supplied/user-extensible]
New Device Support
“Supported” Instruments
“New” Equipment
Shared/Provided
Required
Optional

Taken from the introduction course into EPICS

ACE: alternative DAQ software candidate

- ADAPTIVE Communication Environment
- ACE is a free OO C++ toolkit, including reusable wrappers, classes and network programming frameworks, middlewares, which is portable & supportable in many Operation Systems.
- An off-the-shelf commercial components: Supported commercially by www.riverace.com
- ACE is used by: Boeing, Avionics, Telecom gateway (Ericsson, Kodak, Lucent, Motorola & Siemens…), Electronic medical imaging, etc.

ACE Architecture



The frameworks and patterns layer

The C++ wrapper facade layer

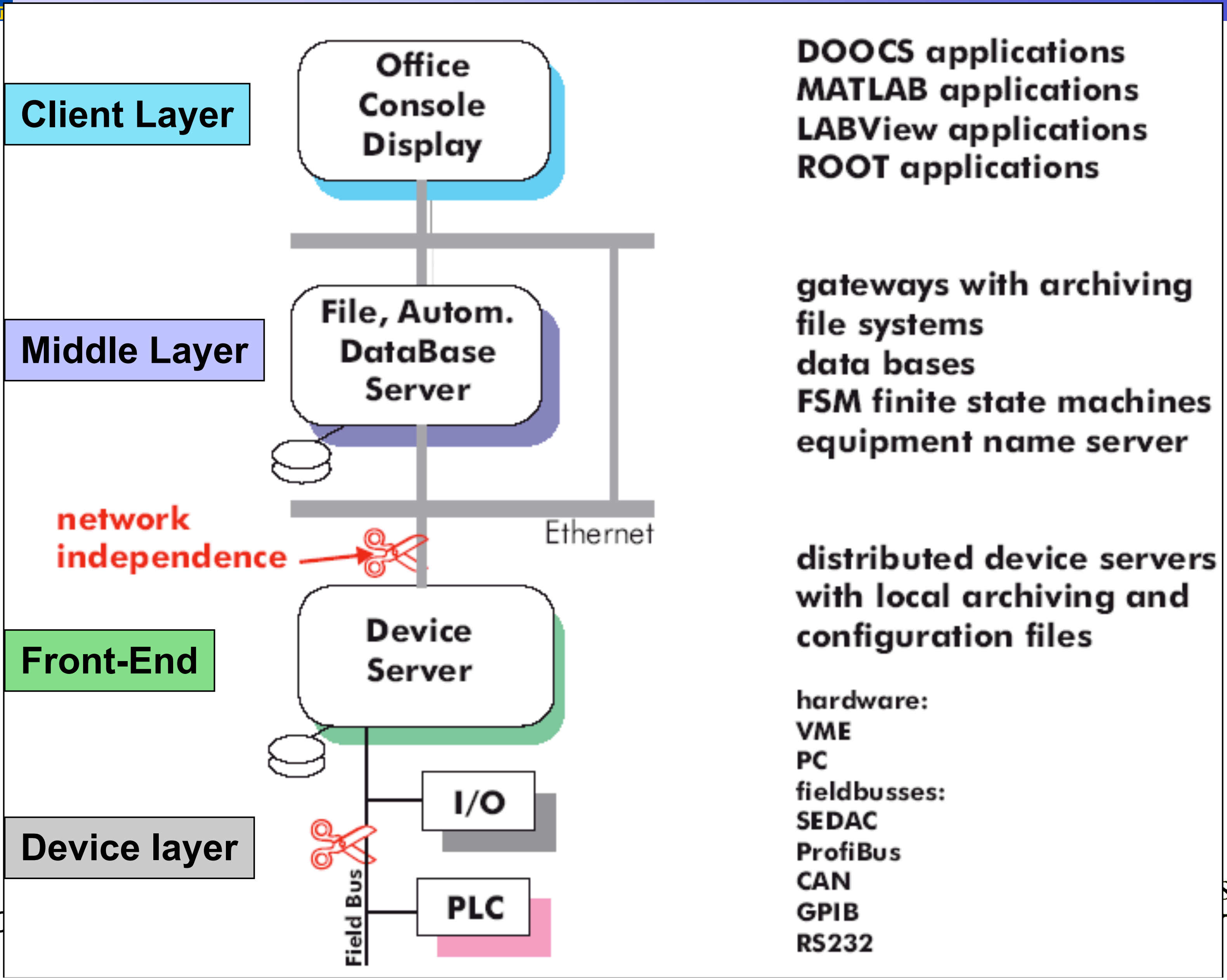
Operating system (OS) adaptation layer

What’s DOOCS?

- **D**istributed **O**bject **O**riented **C**ontrol **S**ystem
- Designed for TESLA Test Facility (TTF), used by HERA and FEL
- Whole system in C++ language
- Standalone control system, allow uniform access to all TTF control system
- Class libs for device server, communication objects and display components.
- The architecture based on OO API on the client side with multiple protocols.

http://tesla.desy.de/doocs/doocs.html

DOOCS Architecture



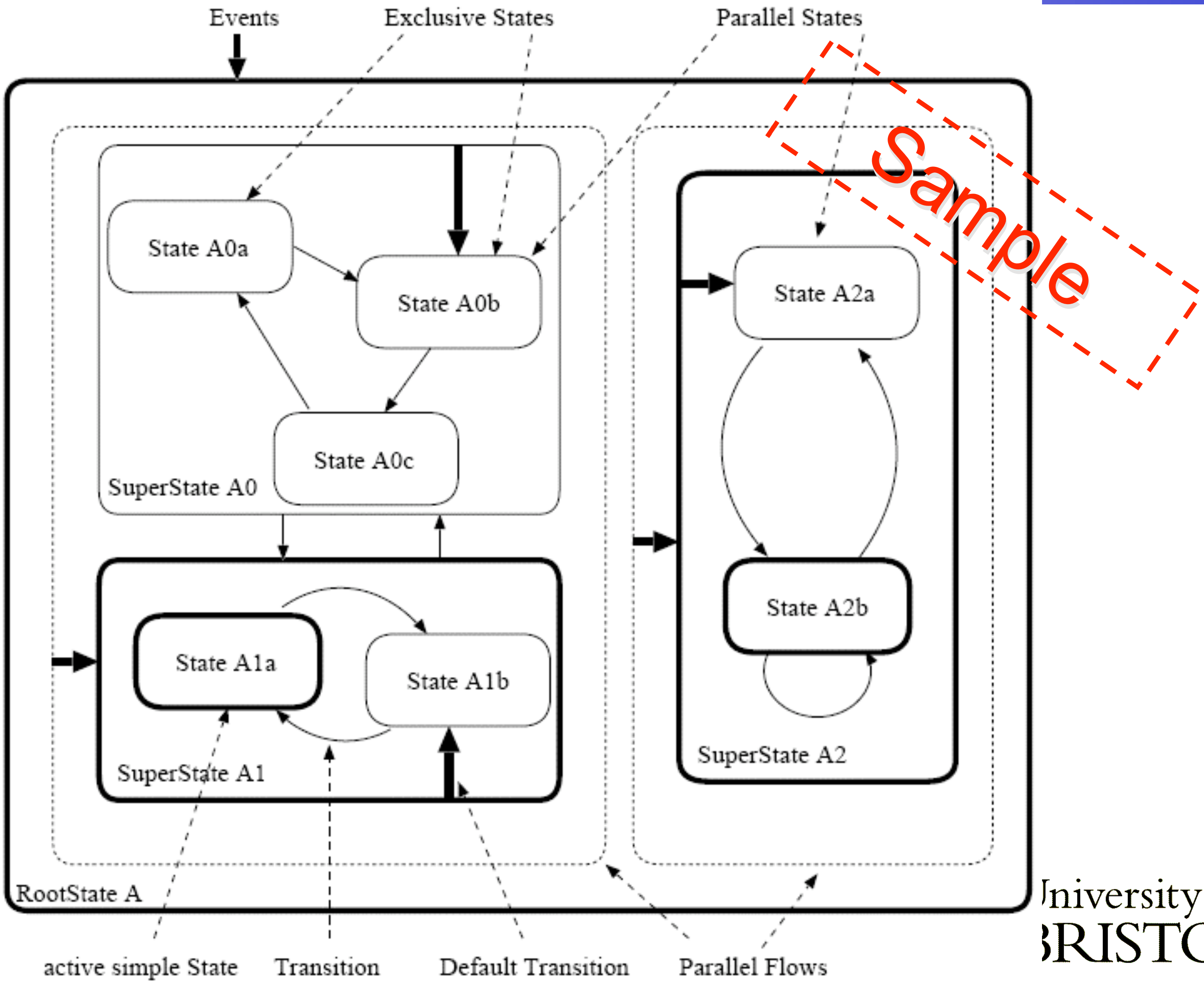
DOOCS applications
MATLAB applications
LABView applications
ROOT applications

gateways with archiving
file systems
data bases
FSM finite state machines
equipment name server

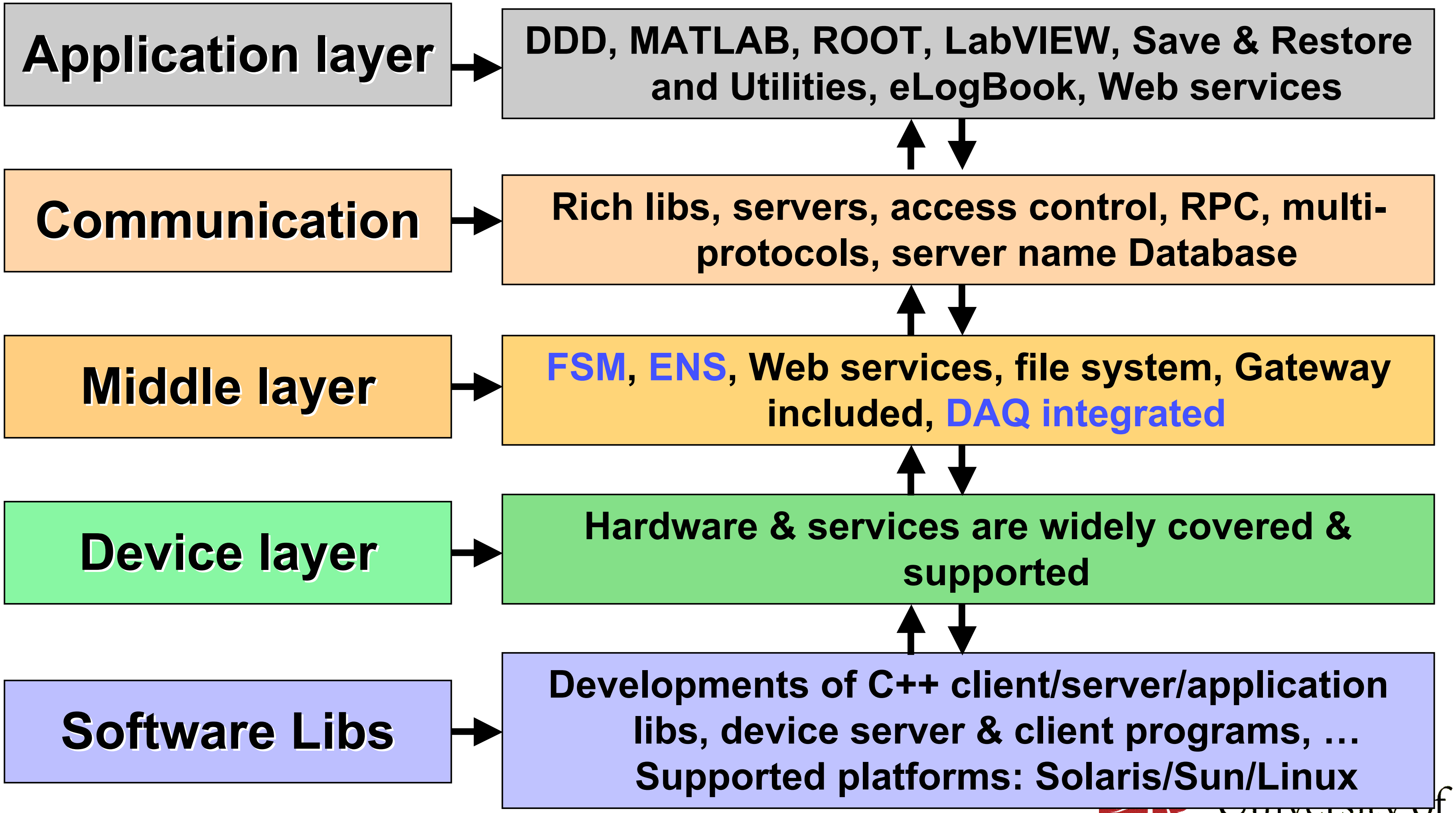
distributed device servers with local archiving and configuration files

hardware:
VME
PC
fieldbusses:
SEDAC
ProfiBus
CAN
GPIB
RS232

Finite State Machine



Main Features of DOOCS



Application layer	DDD, MATLAB, ROOT, LabVIEW, Save & Restore and Utilities, eLogBook, Web services
Communication	Rich libs, servers, access control, RPC, multi-protocols, server name Database
Middle layer	FSM, ENS, Web services, file system, Gateway included, DAQ integrated
Device layer	Hardware & services are widely covered & supported
Software Libs	Developments of C++ client/server/application libs, device server & client programs, … Supported platforms: Solaris/Sun/Linux

JRA3 Framework Review

- DAQ software tasks are reviewed.
- Use cases of DAQ software for EUDET are discussed in some conceptions.
- DAQ software candidates are discussed: EPICS, ACE and DOOCS
- Some comparisons of functionalities are made between EPICS, ACE, DOOCS and DAQ needs.
- Open discussions of EUDET DAQ software framework? EPICS/ACE? DOOCS is more suitable.

Summary

- All working groups got off to a rapid start on DAQ to support activities
- Calice has beam-test DAQ for several test-beam campaigns. JRA3 benefited from this.
    - JRA3 Performing review of DAQ frameworks on the market
- JRA1 has written a light-weight DAQ, used in beam-tests summer '07
- JRA2 using many Alice components. In a position to make rapid progress.