Review of DAQ Software

Tao Wu

Royal Holloway, Univ. of London

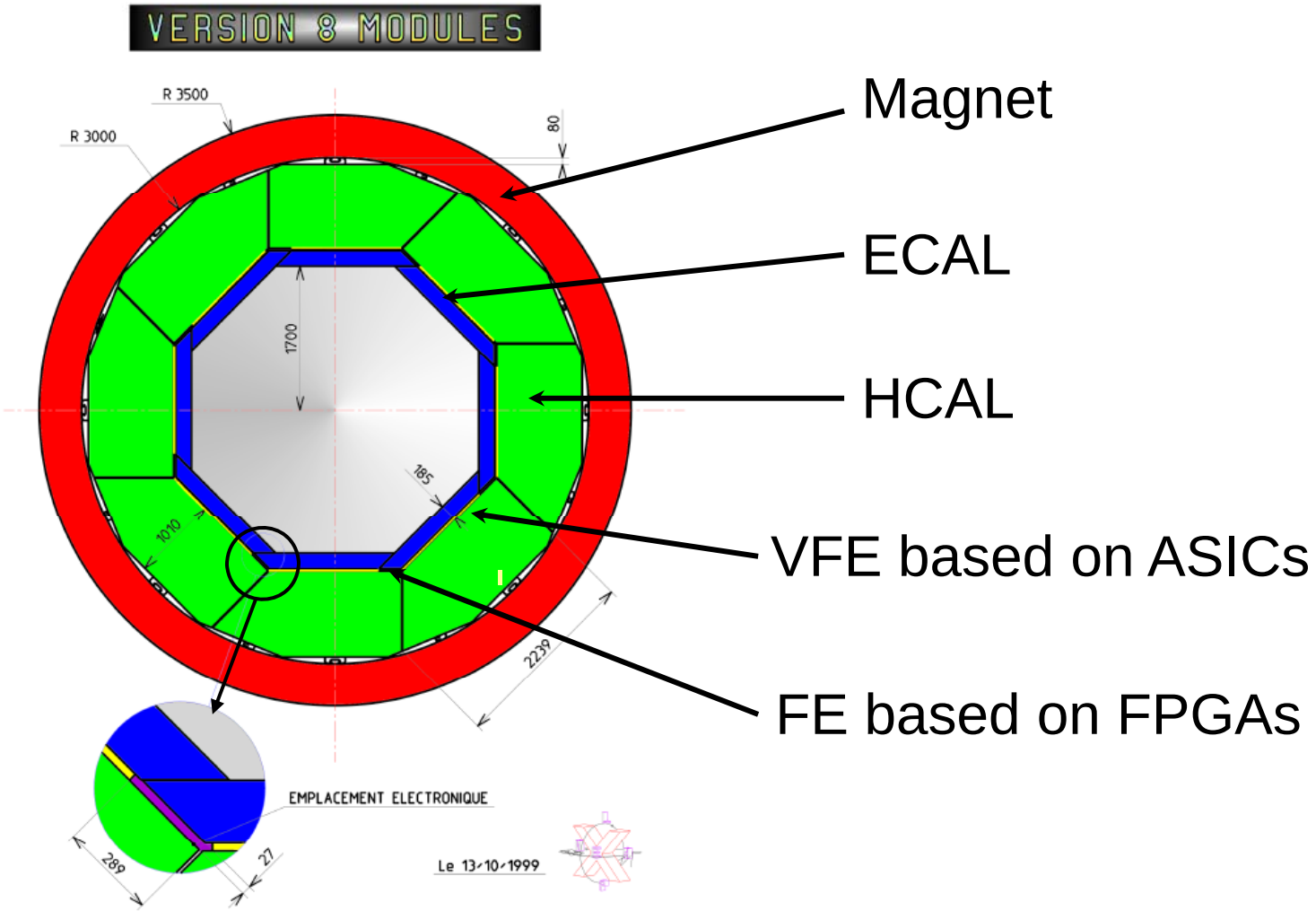
EUDET Annual Meeting @ Paris

08-10/OCT/2007

Outline

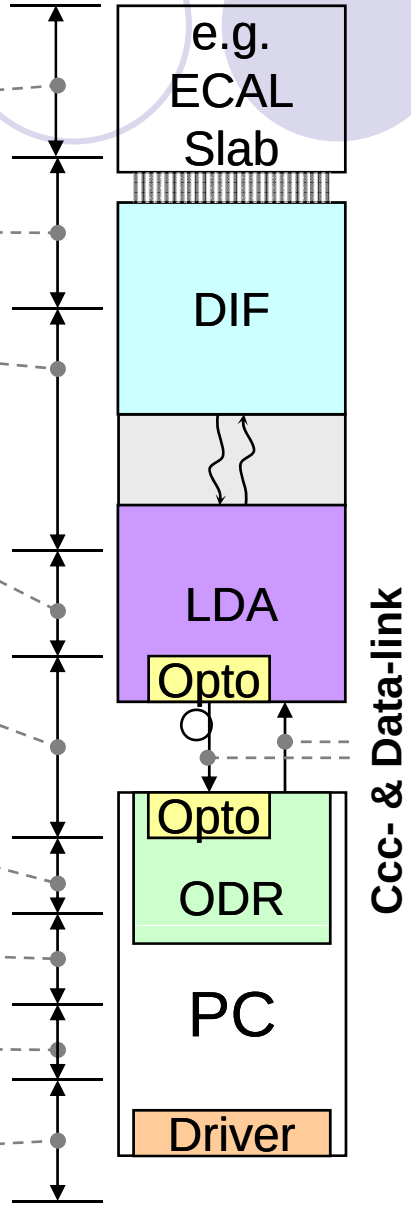
- DAQ overview and software tasks
- Use cases of DAQ software for EUDET
- Discussions of DAQ software candidates: EPICS, ACE and DOOCS
- Summary

Detector Layout



Current DAQ Architecture

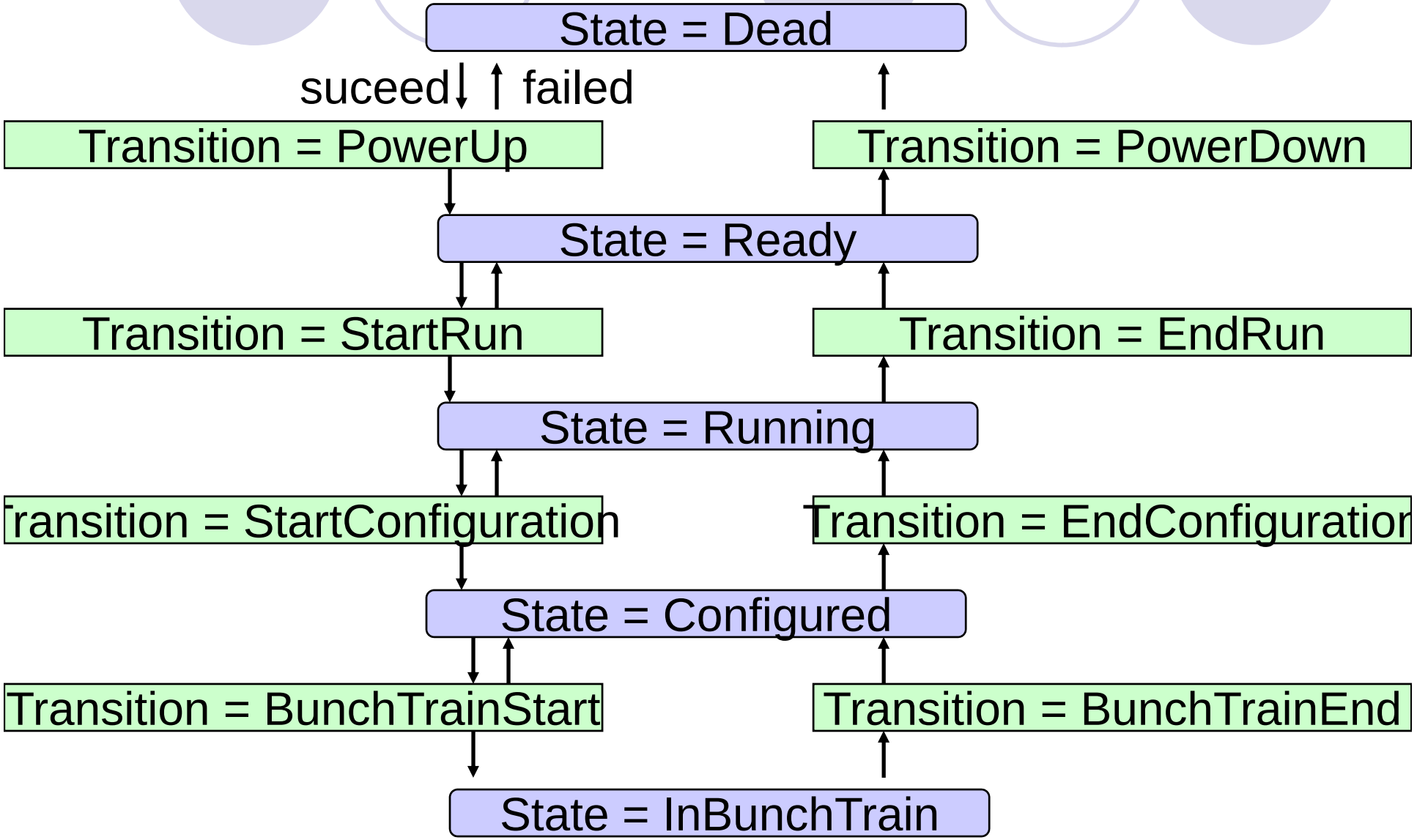
- Slab hosts VFE chips
- DIF connected to Slab
- LDA servicing DIFs
- Link/Data Aggregator (LDA)
- LDAs read out by ODR via opto-links
- Off-Detector Receiver (ODR)
- PC hosts ODR
  - PCI-Express driver software
- Local Software DAQ
- Full blown Software DAQ



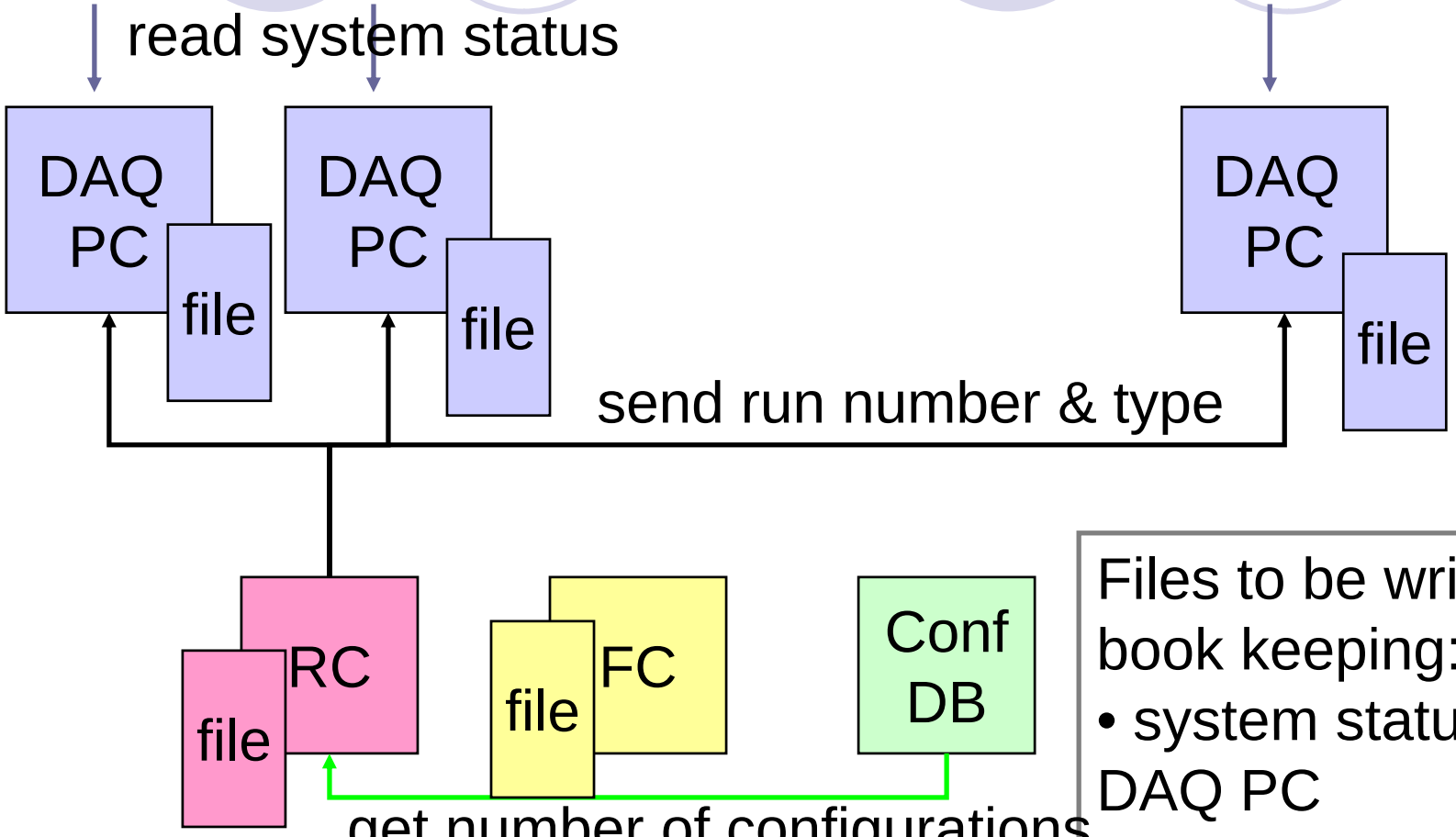
DAQ software tasks

- Aim to develop a generic system
- Maximize use of off-the-shelf commercial components, cheap, scalable and maintainable
- Provide well defined interfaces between DAQ components to allow for simple upgrading or replacement in future without major re-design or cost
- Software control to integrate the rest of sub-systems of detectors
- Software to build event from bunch train data and disparate sources into single event data
- Manage network and data storage

DAQ software for Eudet: State Analysis



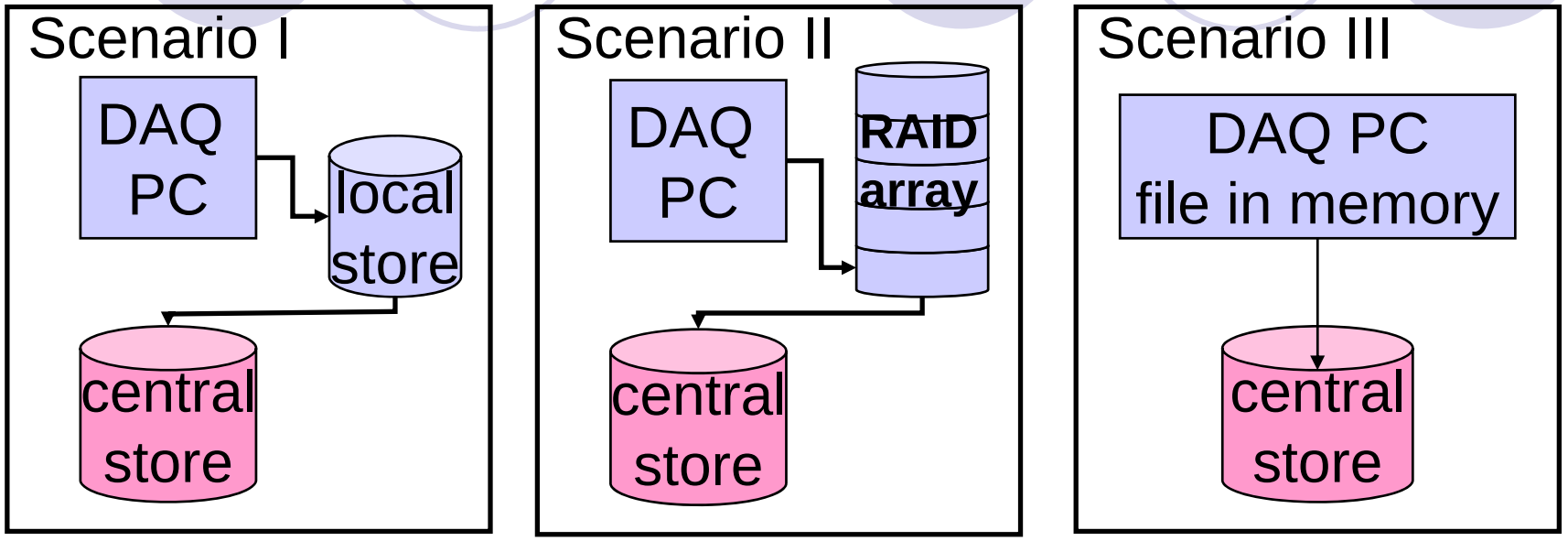
DAQ software for EUDET: Transition: StartRun



Files to be written for book keeping:

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

DAQ software for EUDET: Data Storage



- Which scenario to choose depending on the bandwidth with which the data gets produced: (I) up to 200Mbit/sec, (II) up to ~1600Mbit/sec, (III) from there on
- desirable to have files because transfer is easier and in case of timing problems error handling is easier, but keep system flexible for now

What DAQ software should be used?

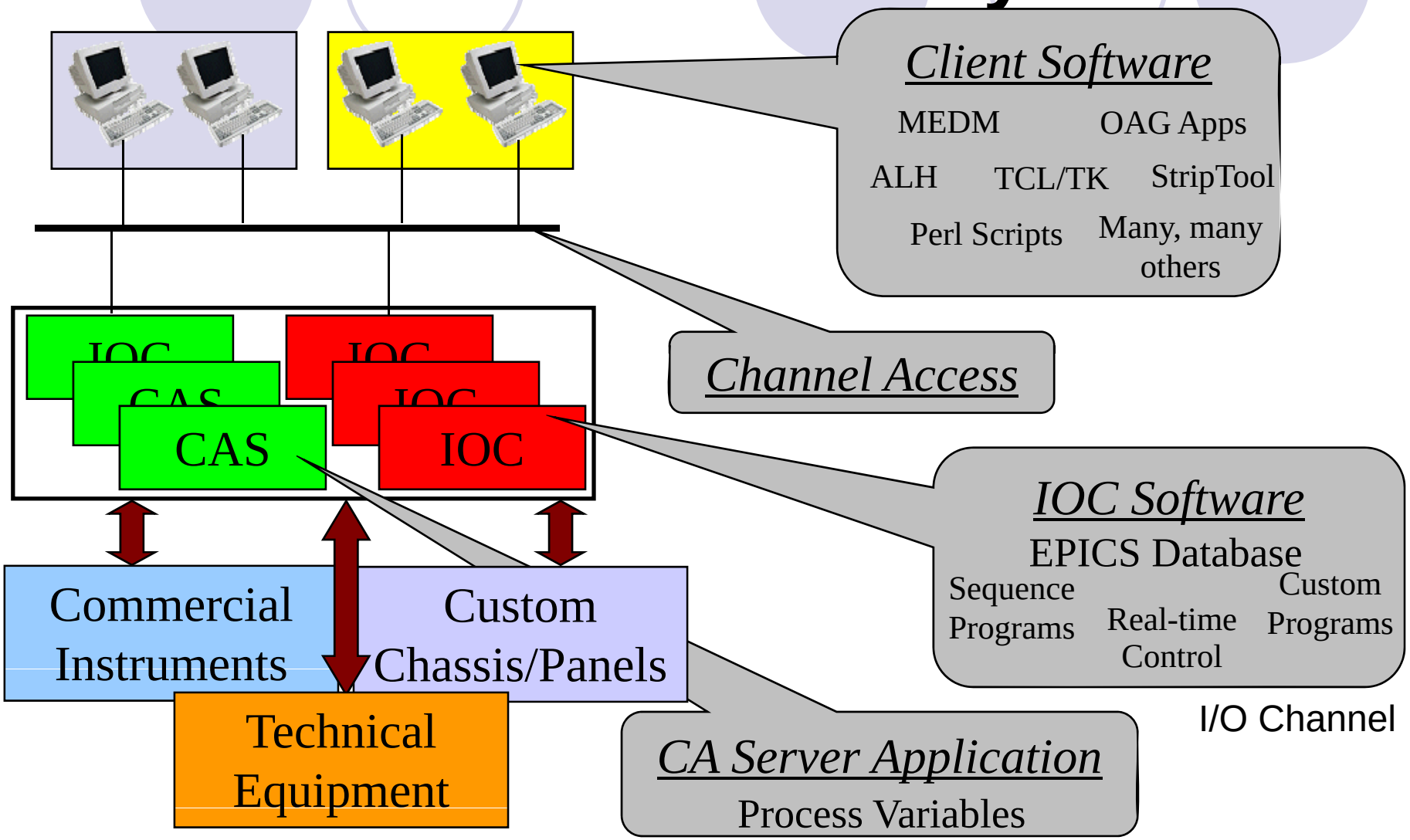
- An effort is focused on EPICS;
- An alternative candidate is ACE;
- Looking into DOOCS framework.

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 …

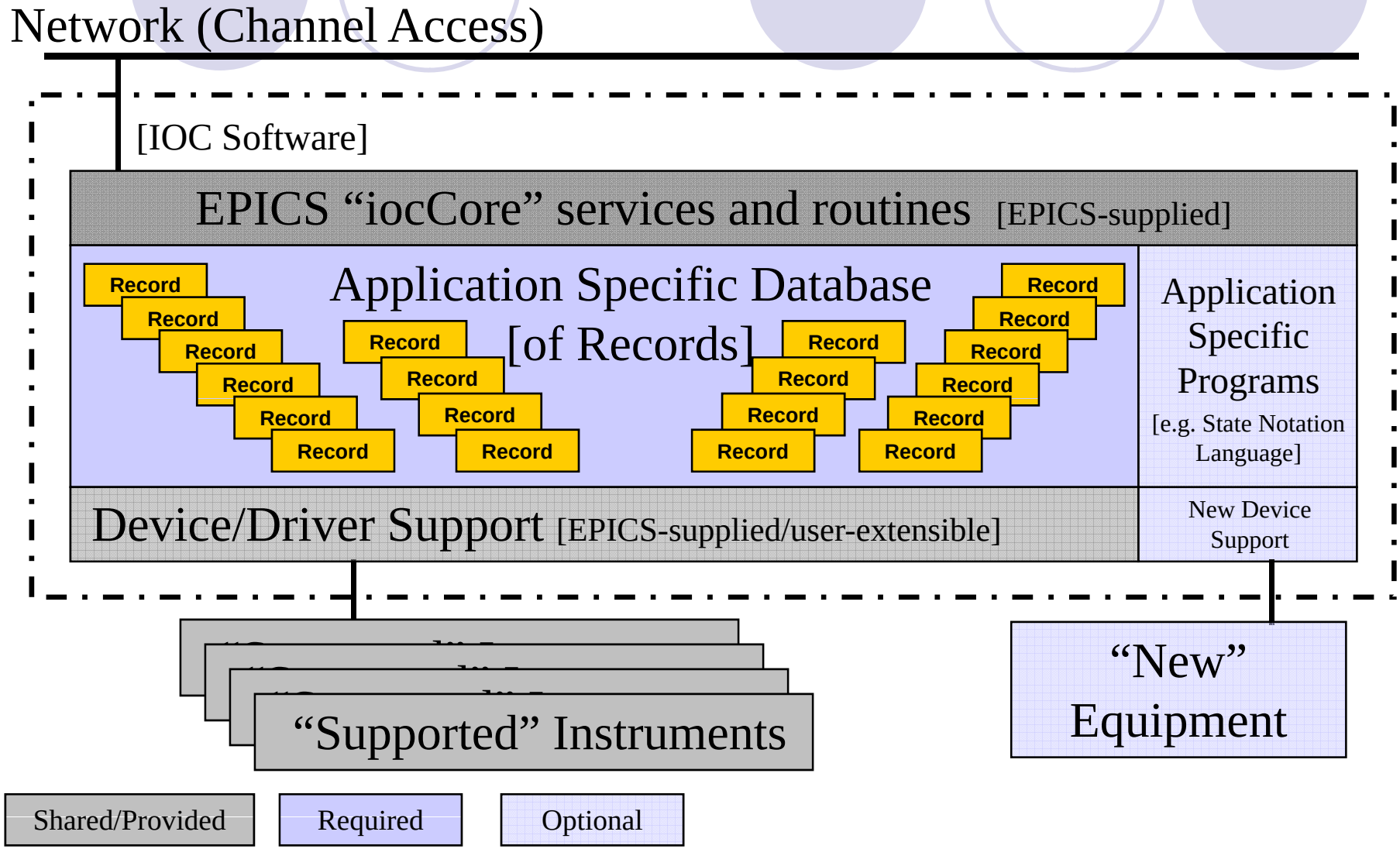
http://www.aps.anl.gov/epics/

Canonical Form of an EPICS Control System



Taken from the introduction course into EPICS

EPICS IOC Software in One Slide



Taken from the introduction course into EPICS

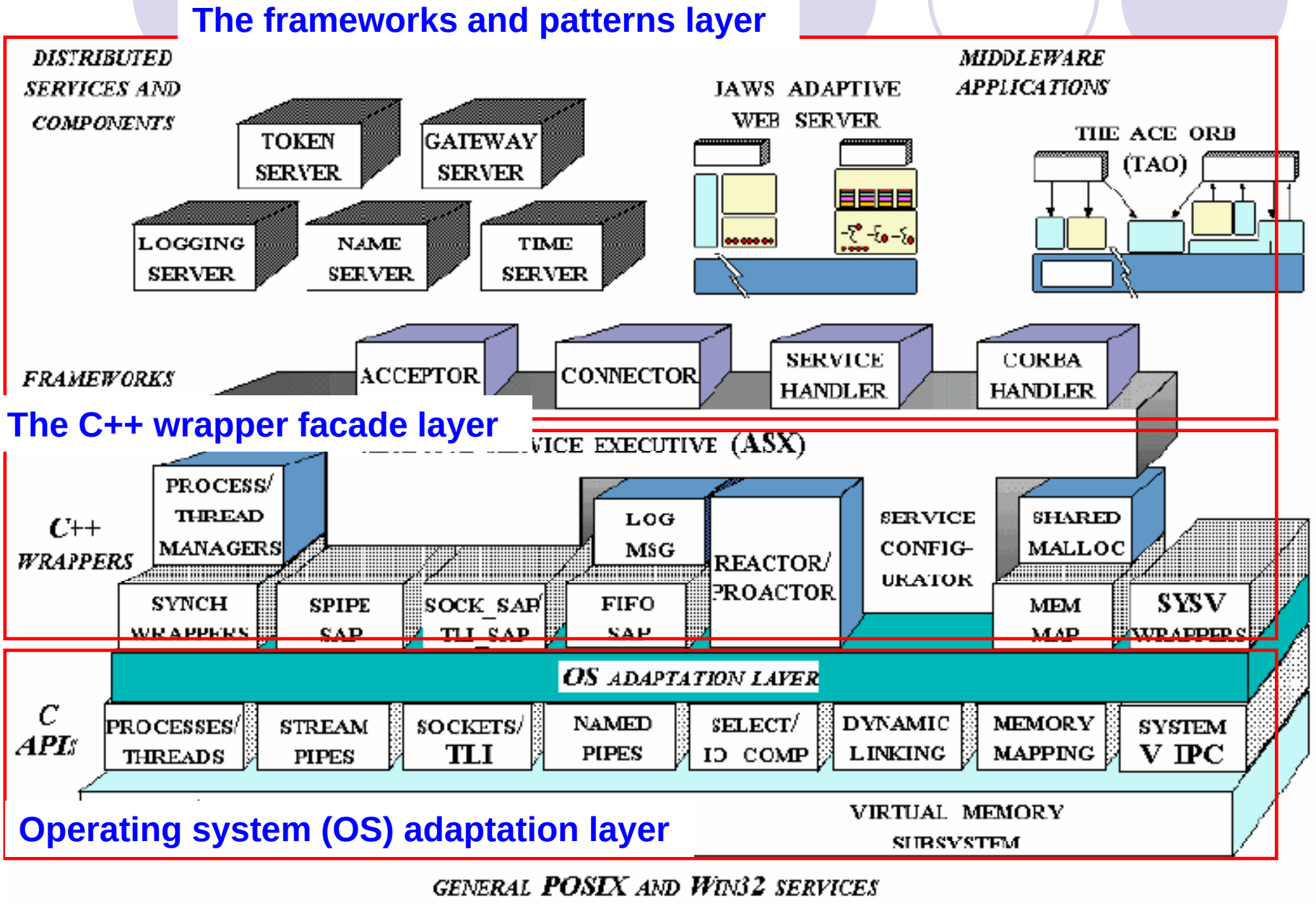
Main features linked to CALICE-DAQ

- Network-based “client/server” model with Channel Access Protocol
- Rich Client Software & Channel Access Server Application and I/O Channel software
- Toolkits: Commercial Instruments, Custom Chassis/Panels and Technical Equipment
- Common uses
  - Provide automated start-up sequences
  - Provide fault recovery or transition to a safe state
  - Provide automatic calibration of equipment
  - Benefit from Run Control and record management

ACE: alternative DAQ software candidate

- **A**DAPTIVE **C**ommunication **E**nvironment
- 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



Main Functionalities of ACE

- **High-performance & real-time communication software**
  - Object Orient network services & applications
  - Services of interprocess communication, event demultiplexing, explicit dynamic linking, and concurrency
  - Automated system re/configuration by dynamically linking services
  - Execute services in one or more processes or threads

- ACE basics: Installation, Logging Facility, Containers
- Interprocess Communication: Sockets, Reactor, Proactor, Other IPC Types
- Process and Thread Management: Process, Signals, Thread, Thread Safety and Synchronization, Tasks and Active Object Pattern, Thread Pools
- Advanced ACE: Memory, Streams, Service Configurator, Acceptor & Connector, Naming Service, Message Queues

ACE functionality vs CALICE DAQ

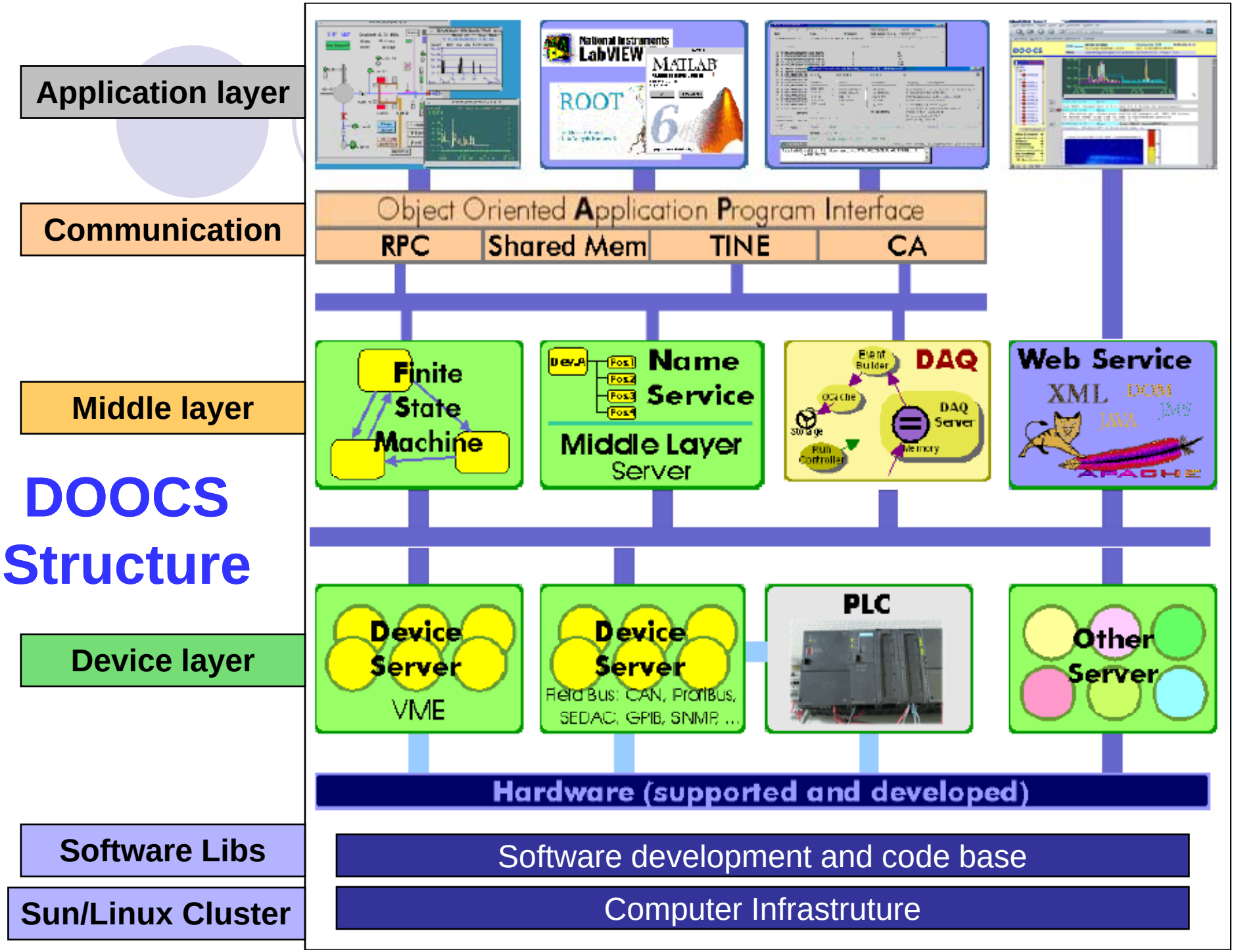
DAQ software for EUDET	ACE
Transition state	Service configurator, message queues
Clock, control	Process, signal, timers
Book-keeping	Logging Facility
Data storage	Memory, stream
Network switch	Acceptor, connector
A/synchronous I/O capabilities	Reactor, proactor
Sub-detector talks	Unicast, broadcast & multi-cast

What's DOOCS?

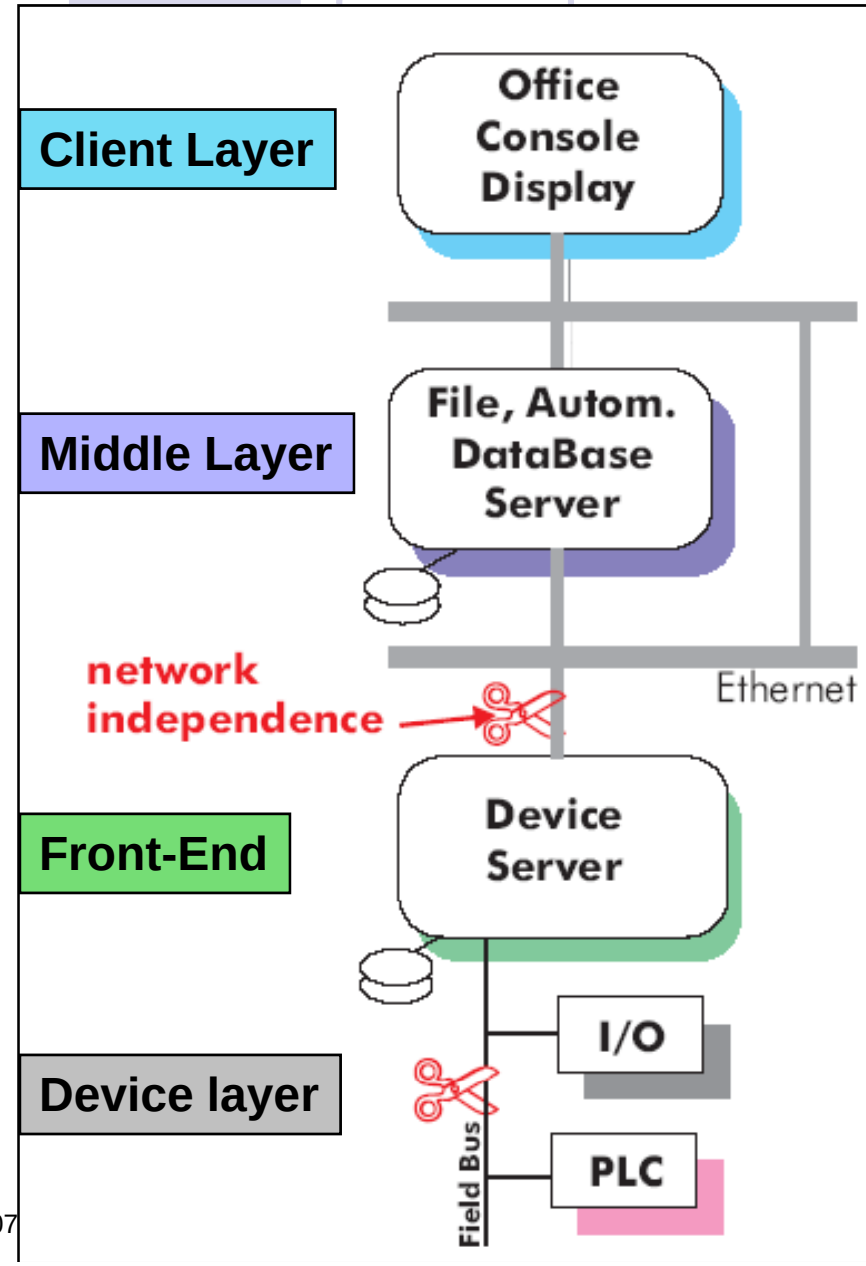
- **D**istributed **O**bject **O**riented **C**ontrol **S**ystem
- Designed for TESLA Test Facility (TTF), used by HERA and FEL
- Whole system in C++ language
- Standalone control system, allow uniform access to all TTF control system
- Class libs for device server, communication objects and display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

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

DOOCS Structure



DOOCS Architecture



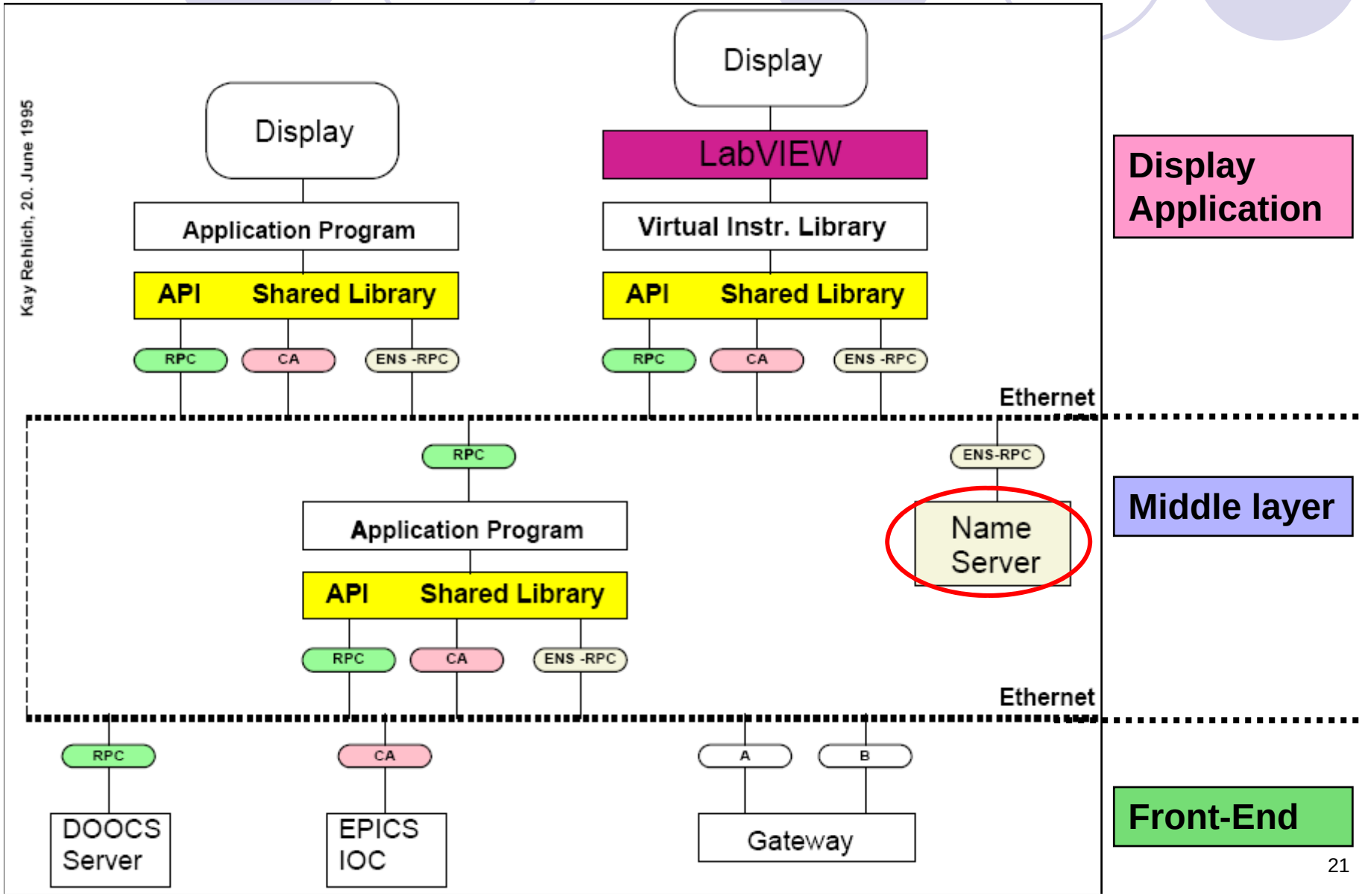
DOOCS applications
MATLAB applications
LABView applications
ROOT applications

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

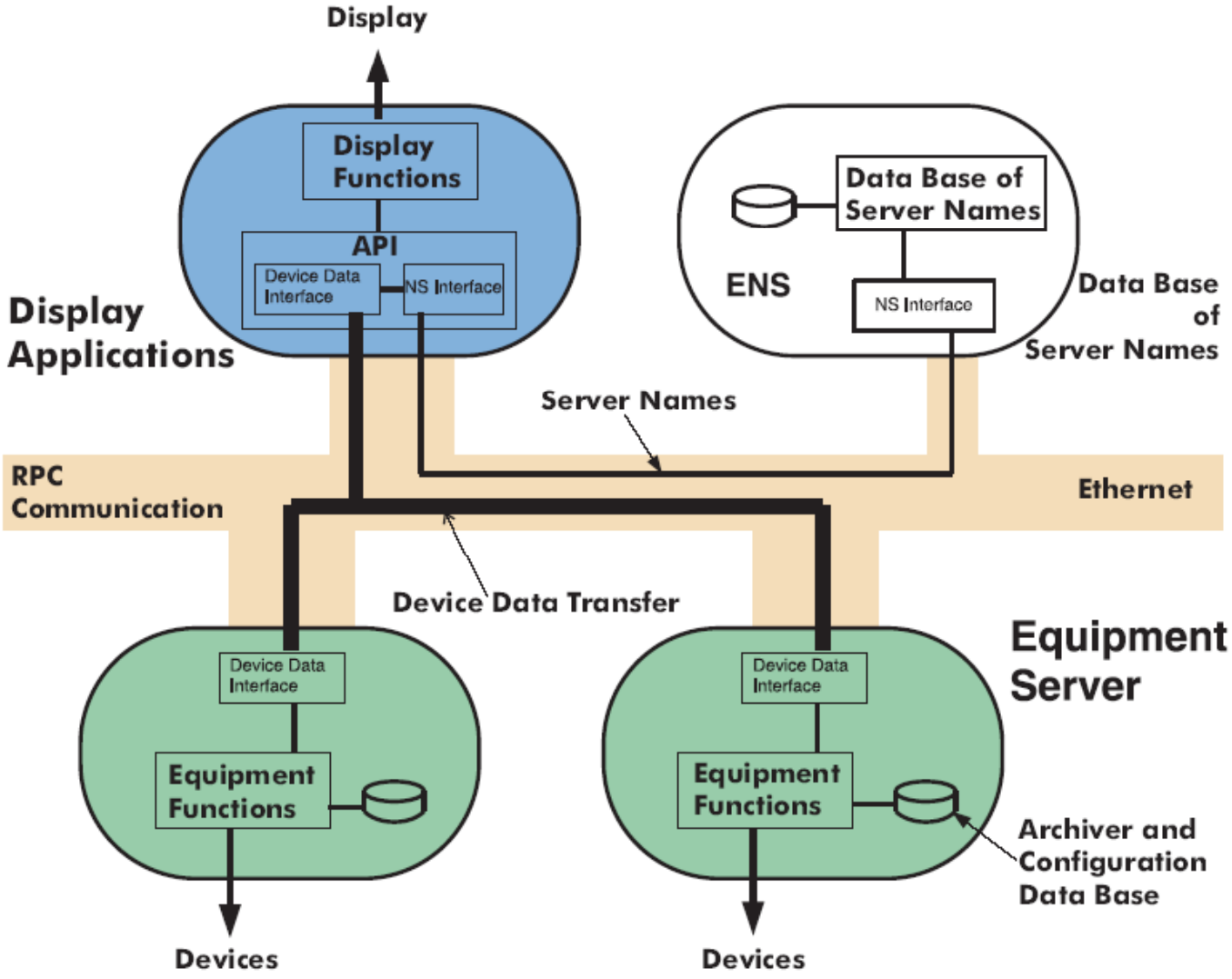
distributed device servers
with local archiving and
configuration files

hardware:
VME
PC
fieldbusses:
SEDAC
ProfiBus
CAN
GPIB
RS232

Software Layers

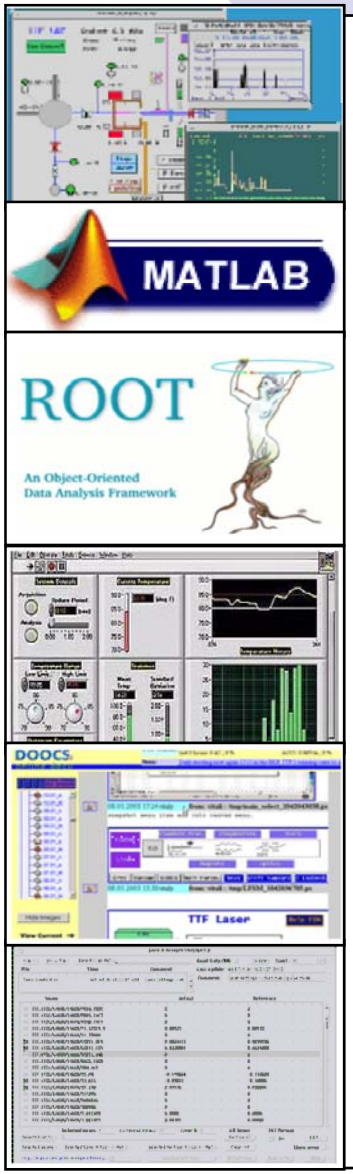


Communication Interface



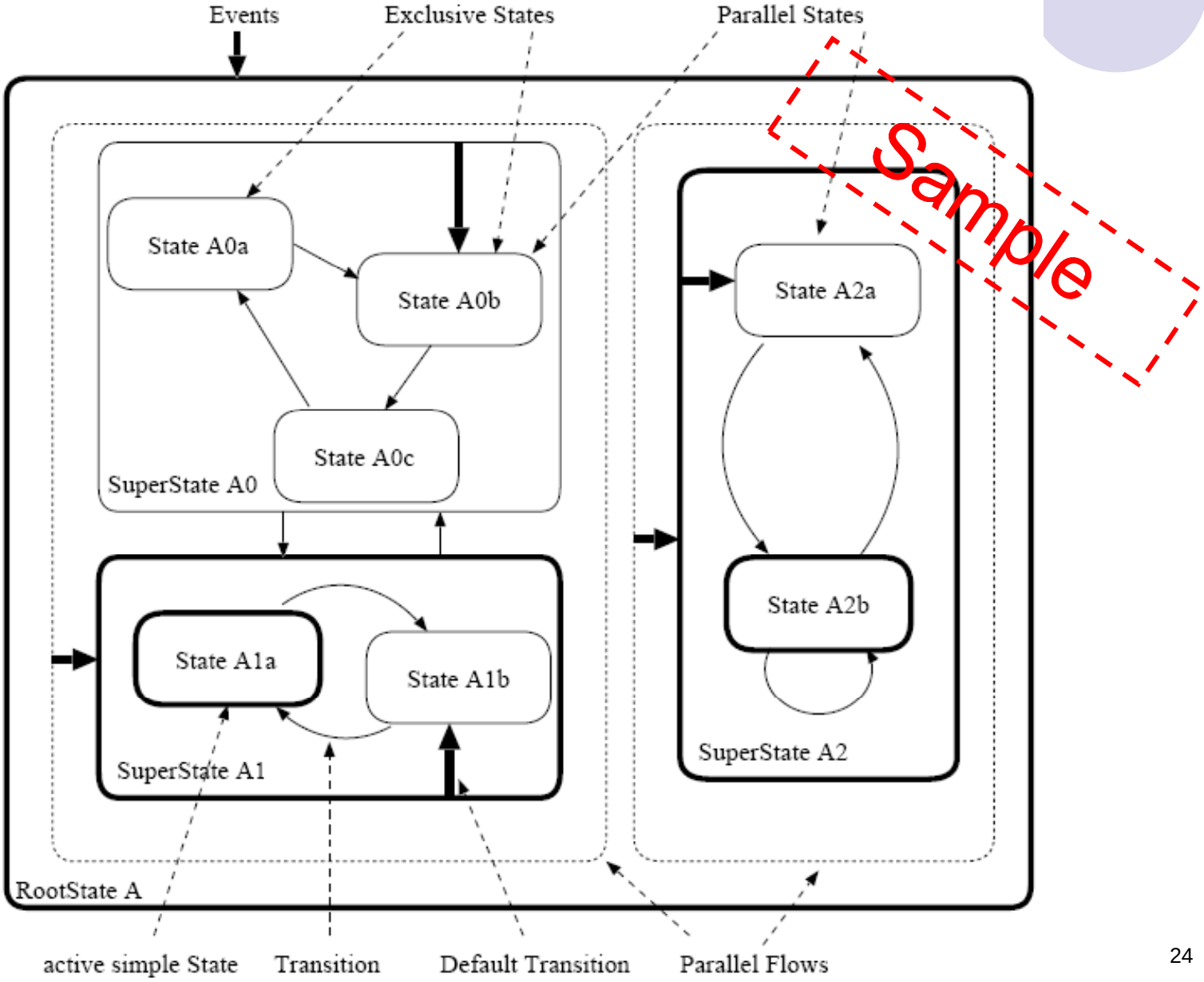
- Distributed System
- Server Name Data Base
- Device Definitions on Server
- Archiving in all Servers
- Based on RPC
- Multiple Protocols
- Variable Data Objects
- Access Control
- C++
- Rich Library
- Runs on:
  Solaris/LINUX/(Windows)

External Client Applications



- **DOOCS Data Display:**
  setup & control devices, start all applications
- **MATLAB**: simulate e.g. RF system,
  measurements & ad-hoc applications
- **ROOT:**
  display DAQ system, display & control the orbit, measure the phase
- **LabVIEW**: condition couplers and cavities,
  operate the test stands, operate the OTR system
- **Save & Restore and Utilities**: save & restore linac settings, manage device configureations
- **eLogBook & other Web services:**
  store comments, results, info, error message; display documentations

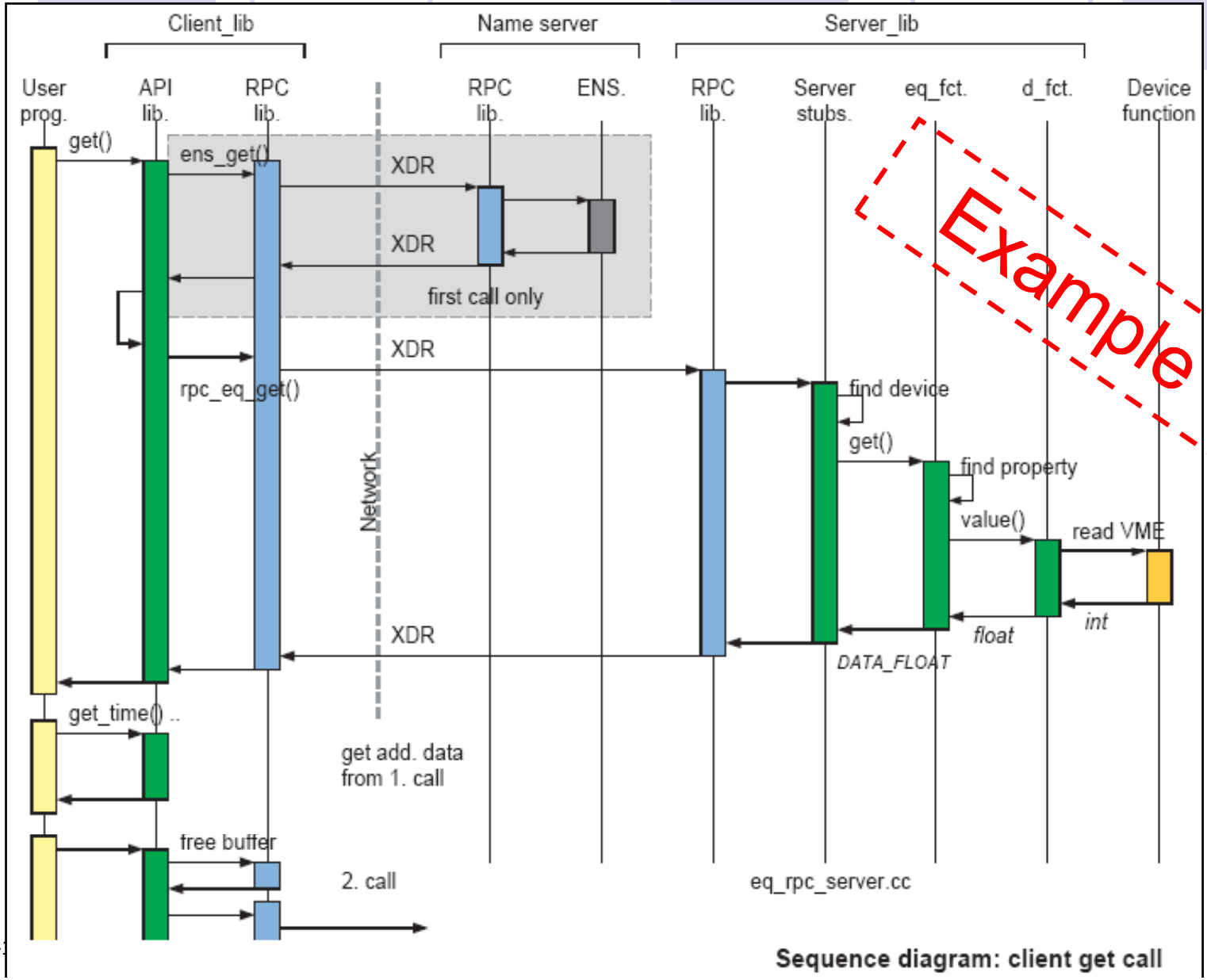
Finite State Machine



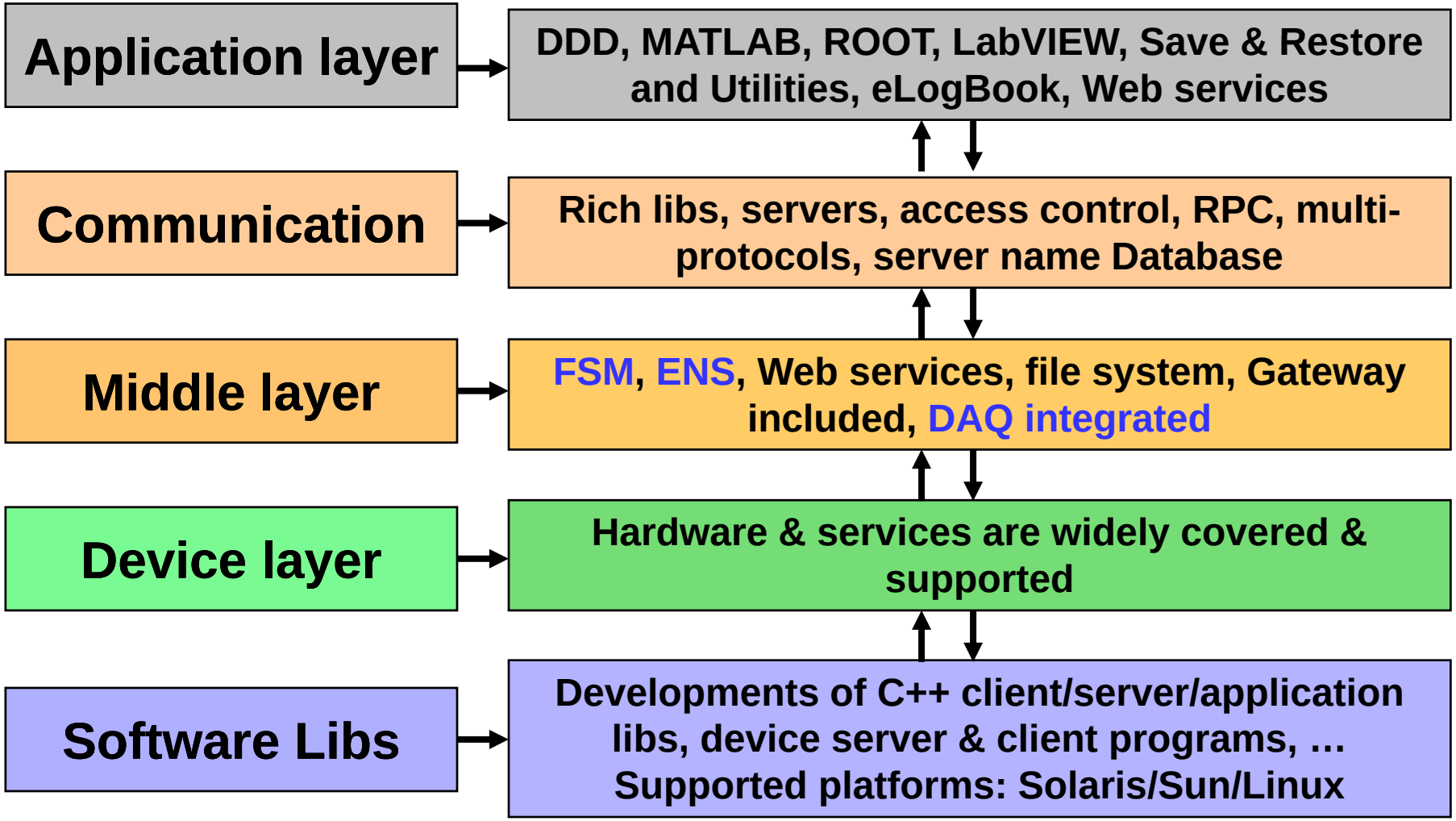
Features of FSM

- FSM is integrated into DOOCS to simplify the automation of operation
- FSM has been integrated into various industrial control system
- Features: highly modular design, graphical design, run-time display, optimized procedures
- Nice FSM concepts and implementation

Sequence of a Client Call



Main Features of DOOCS



Summary

- 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.
- Please help to contribute!

Acknowledgement

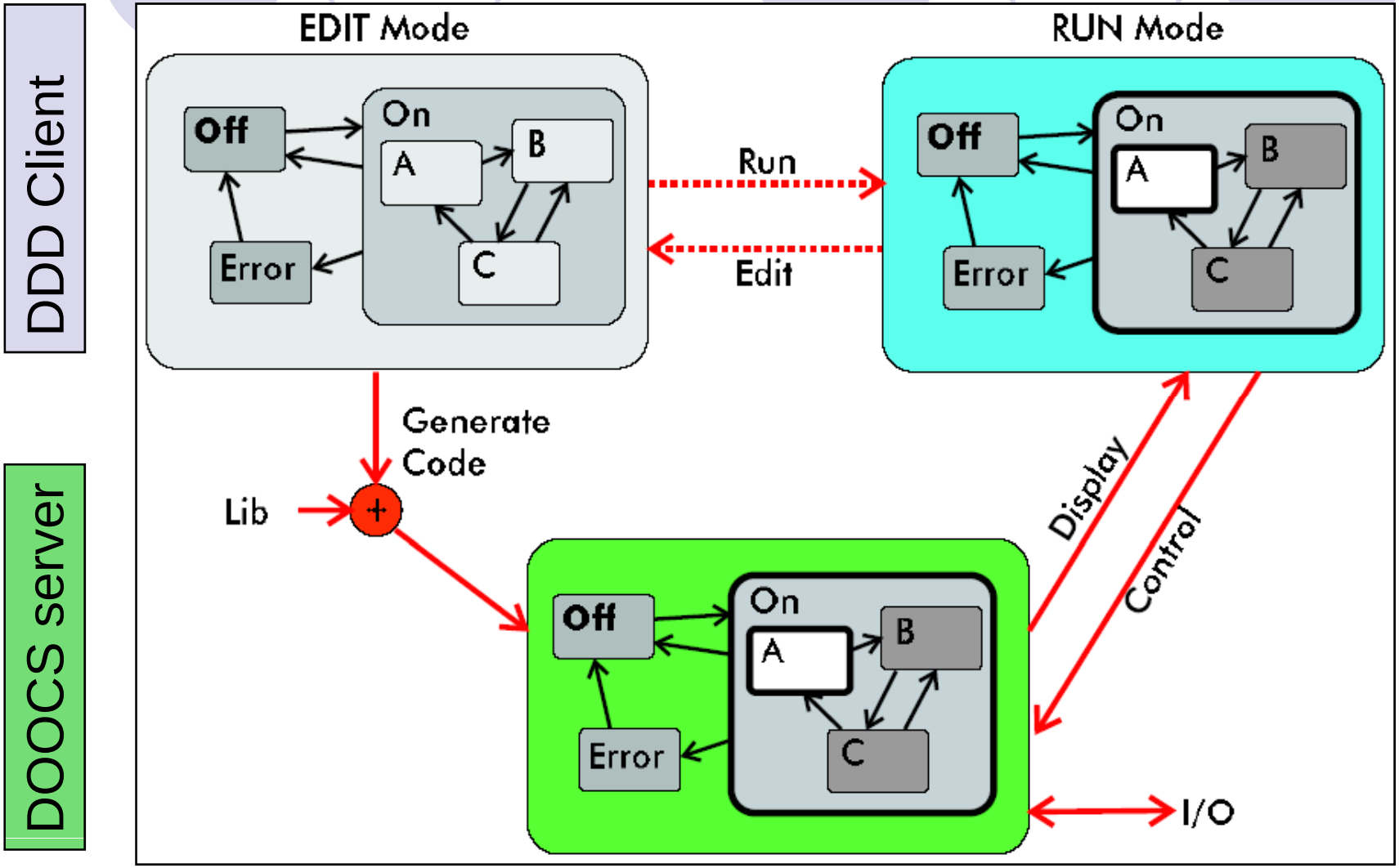
Thanks to CALICE-UK DAQ Collaboration

especially for

- David Bailey (Manchester),
- Paul Dauncey (IC),
- Matthew Wing, Matt Warren,
  Valeria Bartsch (UCL)

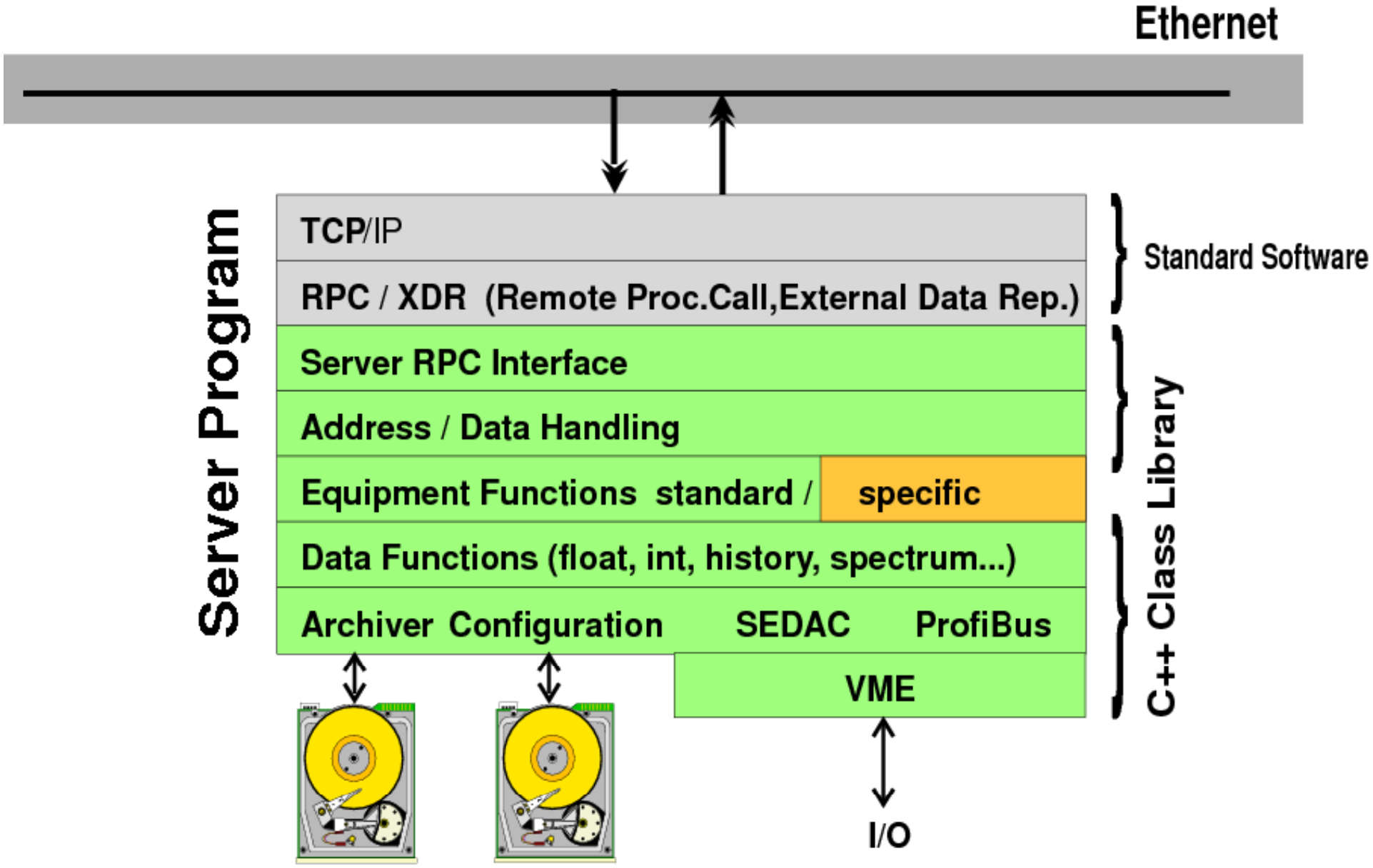
Backup

Finite State Machine



Interaction between client and server side of an FSM in the DOOCS

Device Server



DOOCS server structure

O. Hensler, 4. Jan. 1996