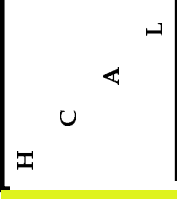
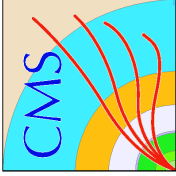


HCAL DCS

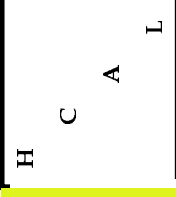


Serguei Sergueev

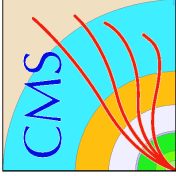
Status and plans for DCS at CERN



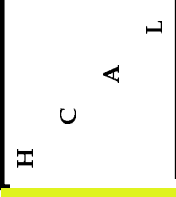
Outline



- **News**
- **Framework**
 - Partitioning + FSM
 - HV control system
 - Parameter downloading
 - SCADA configuring language
- **ELMB**
- **DIM**



Outline (II)

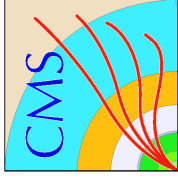


Plans

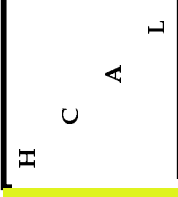
- LV control system
- HCAL tree – urgent

AOB

- Some recommendations
- Kylix

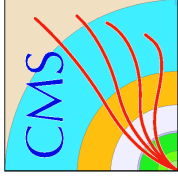


Framework

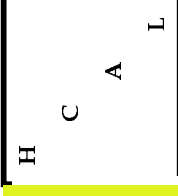


For today the framework includes
following main parts:

- PVSS II
- Set of recommendations and rules
- Partitioning system + FSM
- HV control prototype
- Parameter downloading prototype
- SCADA configuring language
- External alarming system



Partitioning+FSM



Used for the partitioning of the
logical tree

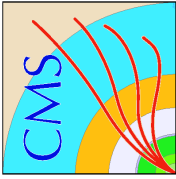
Allows to:

- Propagate command and status messages
- Disconnect and reconnect some part of the logical tree
- Contains the FSM itself

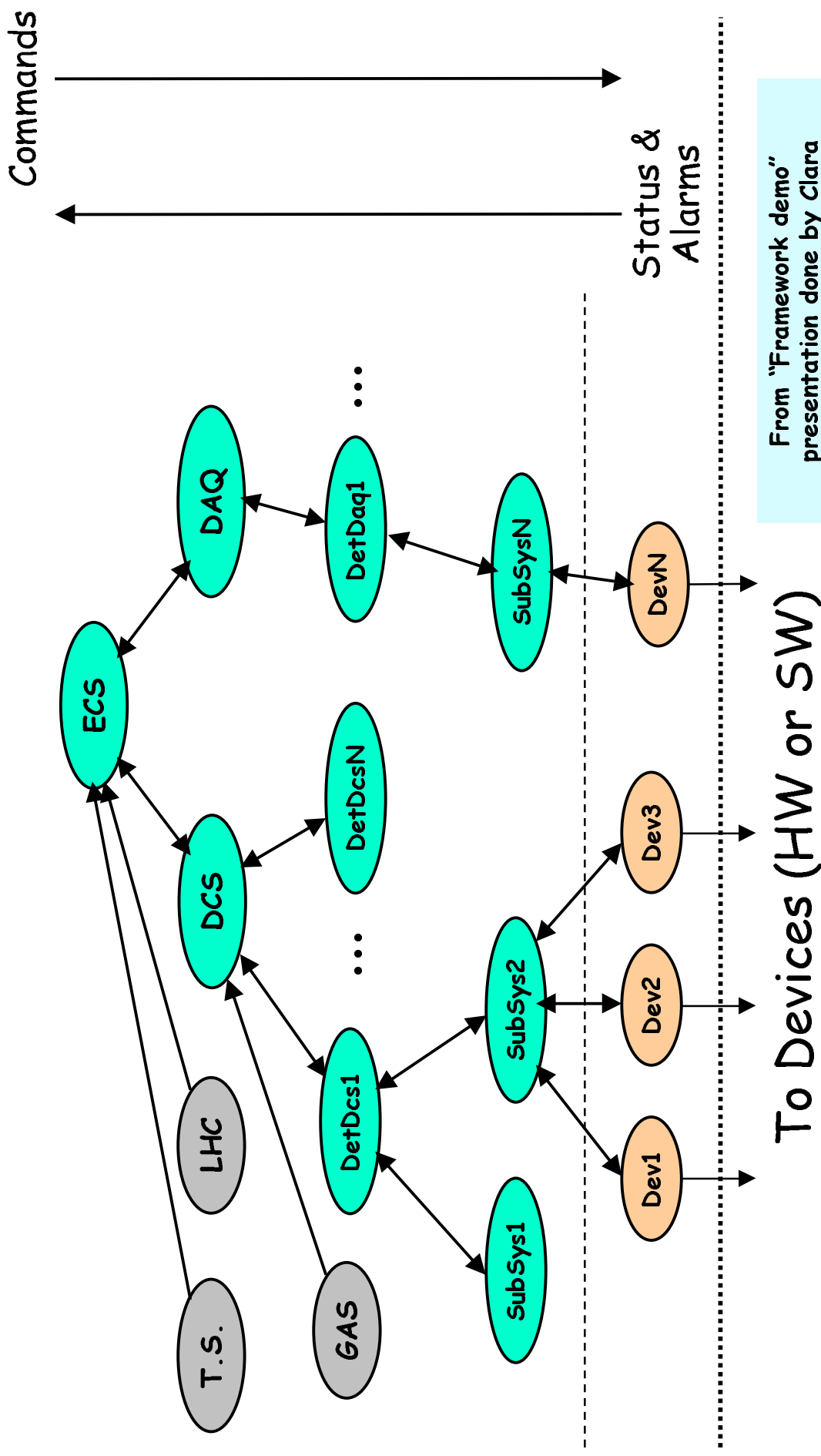
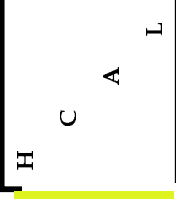
For more details see

<http://itcwww.cern.ch/jcop/projectteam/minutes/2000/Slides/>

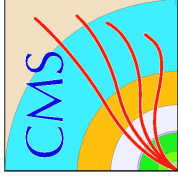
FWDemoJCOP.pdf



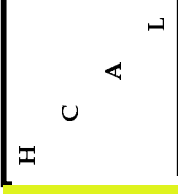
Control System Architecture



From "Framework demo"
presentation done by Clara
Gaspar, EP, CERN



Control System Units



ECS

Control Unit

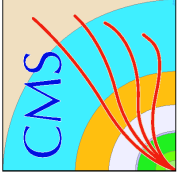
- ✓ Implements Sub-System behaviour (FSM)
- ✓ Implements Partitioning rules
- ✗ Alarm Handling, etc.

Dev.

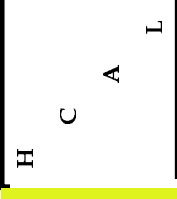
Device Unit

- ✓ "Handles" the device (using SCADA tools)
- ✓ Generates a "State" from device data
- ✓ Implements "Commands" on device
- ✓ Is also a Control Unit

From "Framework demo"
presentation done by Clara
Gaspar, EP, CERN



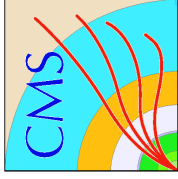
Control unit



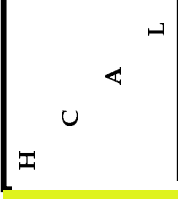
Characteristics:

- **State**
- **Ownership**
 - Only one owner (tree owner)
- **Exclusivity**
 - Exclusive - Only the owner can send commands
 - Shared - Everybody can send commands
- **Mode**
 - Included
 - Excluded
 - Delegated
 - Disabled
 - Ignored

From "Framework demo"
presentation done by Clara
Gaspar, EP, CERN



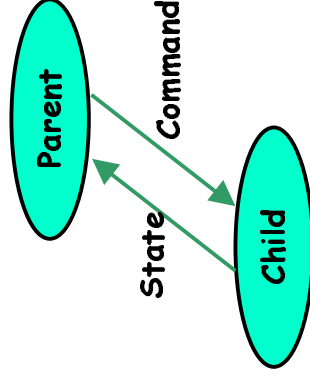
Partitioning Rules



INCLUDED

In a Tree

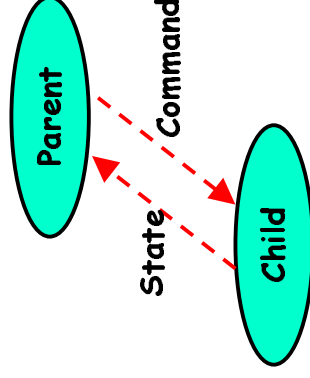
Owner: Tree owner



EXCLUDED

Out of Tree

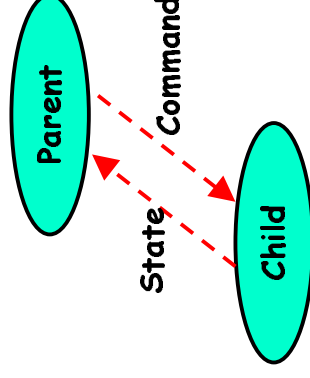
Owner: None



DELEGATED

New Tree (child is root)

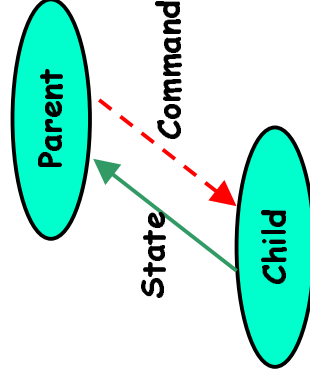
Owner: New tree owner



DISABLED

In a Tree

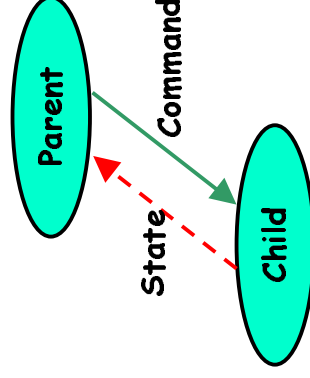
Owner: Tree owner



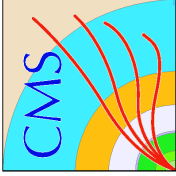
IGNORED

In a Tree

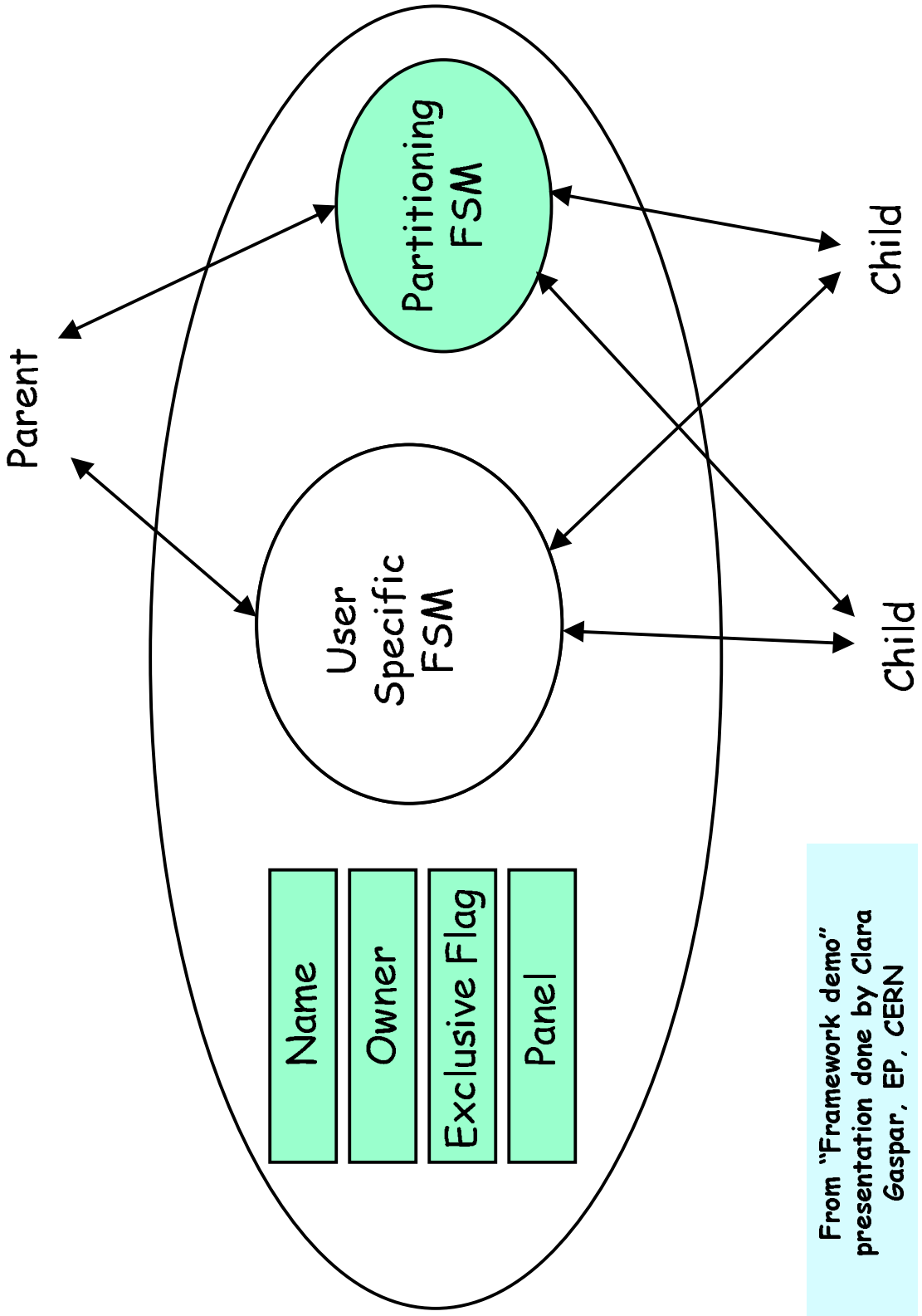
Owner: Tree owner



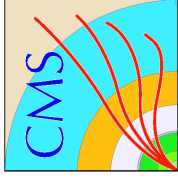
From "Framework demo"
presentation done by Clara
Gaspar, EP, CERN



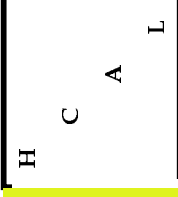
Control Unit Blocks



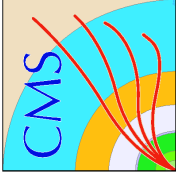
From "Framework demo" presentation done by Clara Gaspar, EP, CERN



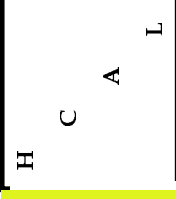
HV control system prototype



- At the present moment under modification to fulfill to the FSM requirements (IT/CO)
- The existing version could be used as the testing and “education” tool



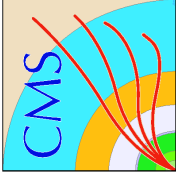
Parameter downloading (I)



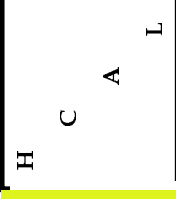
The system consists of set of PVSS panels to create, store and select the “run configuration” .

The parameters could be stored separately from the PVSS in the configuration DB (ORACLE 8).

The FE SW should provide access to the DB and the constants downloading.



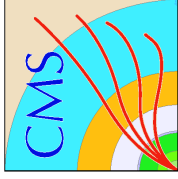
Parameter downloading (II)



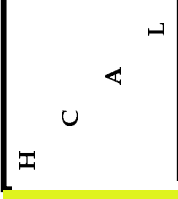
After the “configuration” has been created and adjusted it should be “registered” .

After that moment the “configuration” is registered, it is “locked” and could not be modified any more.

Work in progress...



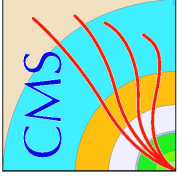
SCADA configuring language



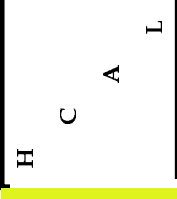
Used for the configuring of the system by using of scripts. Scripts could be automatically generated.

Easily expandable by adding of extra functions to the appropriate directory of the PVSS project

Syntax is similar to Windows .ini files

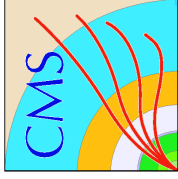


ELMB

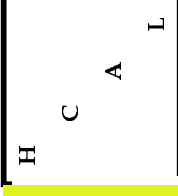


Technical data

- Size 50x66 sq.mm
- CAN bus
- 64 16-bit ADC channels
- 24 digital I/O lines
- 125 Kbaud data exchange rate
- 500 m bus length
- 16 (32) nodes per bus
- Channel refresh frequency – 4..60 Hz
- Could contain embedded programs
- 250 CHF/board



ELMB interface

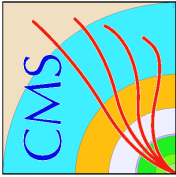


Hardware

- National Instruments CAN-bus interface
- 2 channels/board
- Allows to set many boards/PC

Software

- OPC server (NI toolkit) is available.
Designed at CERN. **Could be used for the RBX interfacing to SCADA.**
- Supports CAN-open protocol
- Negotiation about Linux driver with DIM server



ELMB

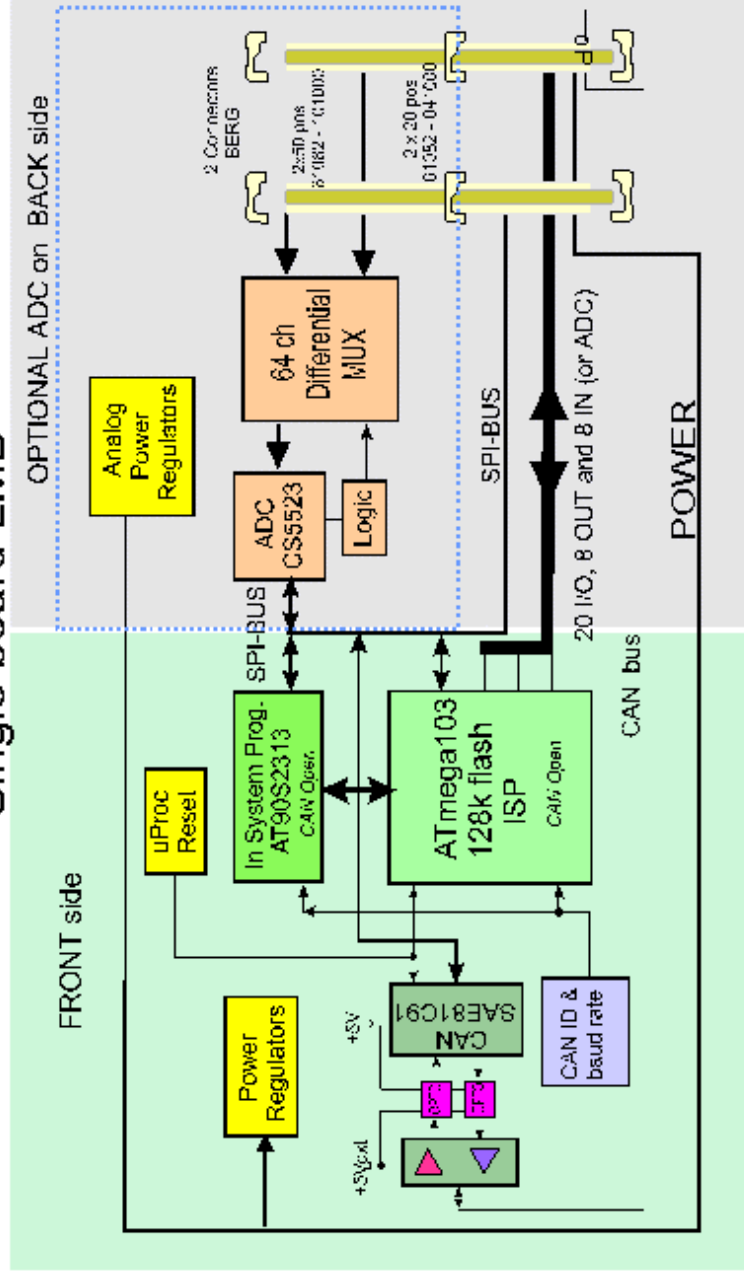
H
C
A
L

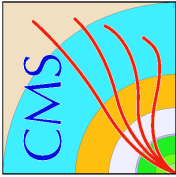


ATLAS
DCS

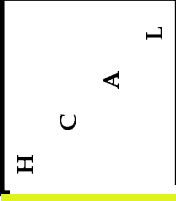
Block Diagram of ELMB

Single board LMB



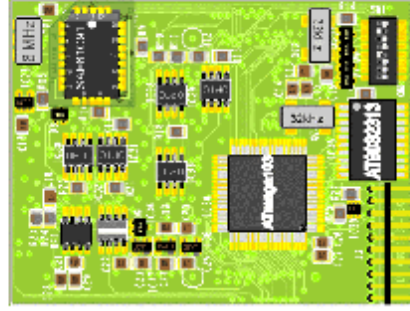


ELMB

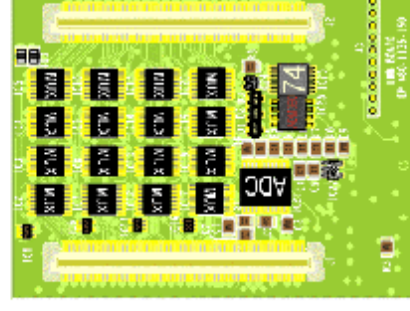


ELMB Layout

LMB single board Side A



Side B



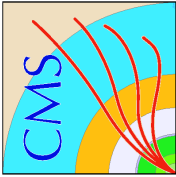
- Processor for CAN protocol
 - User processor 128kb Flash
 - 36 programmable I/O lines
 - digital status
 - digital control
 - DAC
 - local buses (JTAG, I²C)
 - additional memory
 - interrupts
- SPI bus
64 channel ADC 16 bit

Optional ADC on the back side

50x66mm

5mm ↑ ↓ 3.5mm



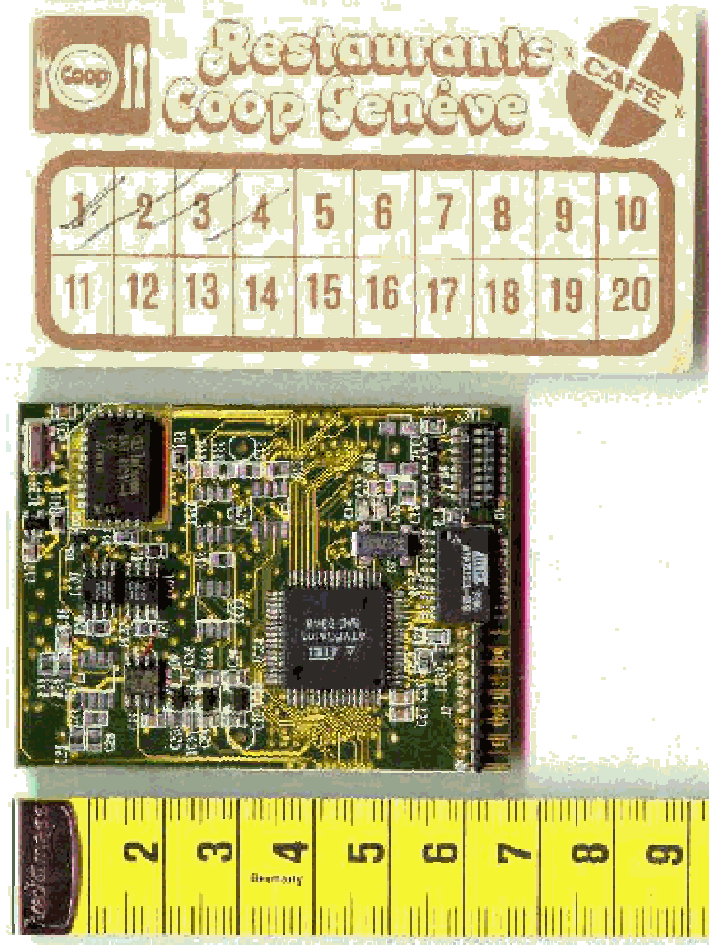


H
C
A
L

ELMB

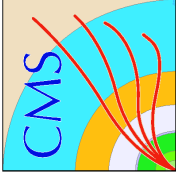


ELMB CAN-side

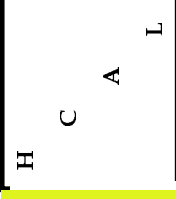


JCOP Meeting 28/6/2000, H.J. Burckhart

18

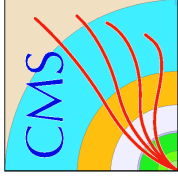


ELMB

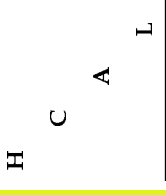


SW Features (cont.)

- ◆ **Common Framework**
 - CAN protocol handling
 - I/O libraries
 - ADC
 - digital I/O lines
 - SPI
 - Example programs
- ◆ **Additional Functionality**
 - Libraries for local buses (e.g. JTAG, I²C)
 - Interrupts
 - Low level loops (e.g. warning/alarm detection)

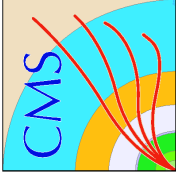


DIM



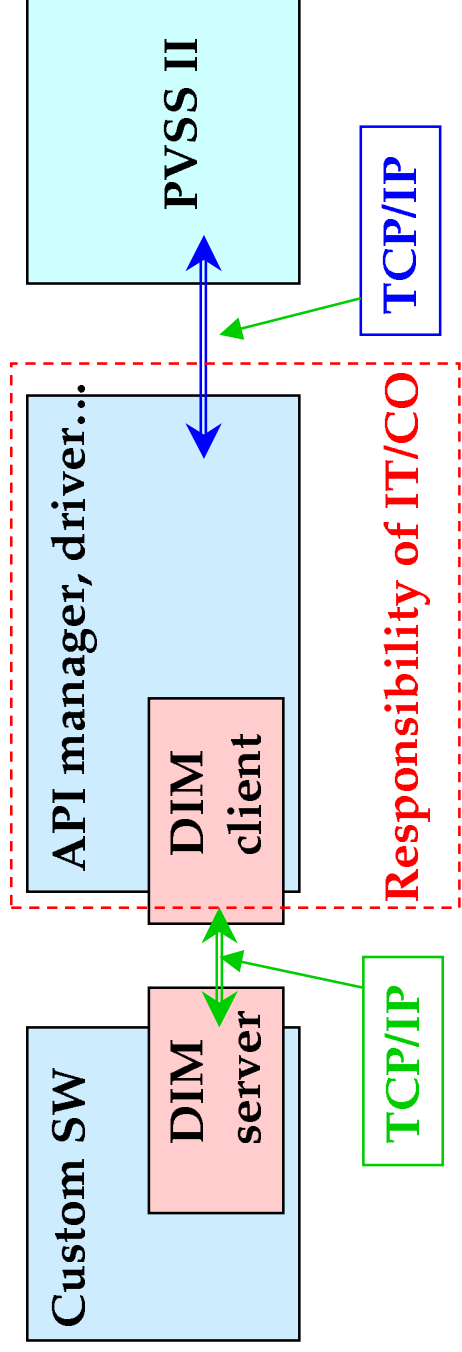
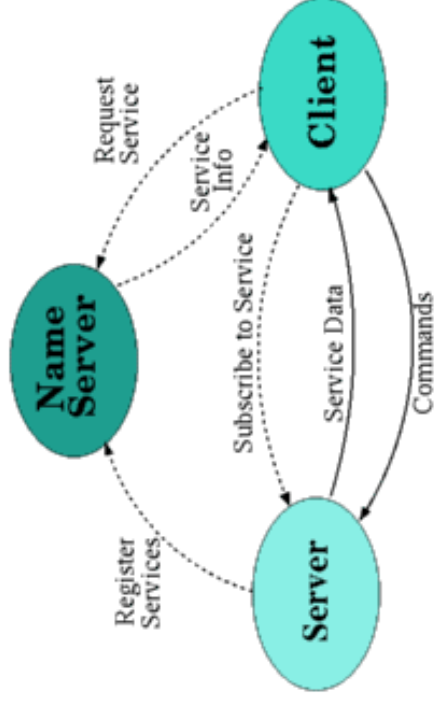
DIM is a communication system for distributed / mixed environments, it provides a network transparent inter-process communication layer.

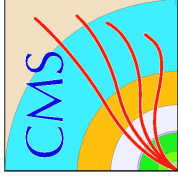
- **Designed at CERN**
- **Used at LEP, ...**
- **Supported by IT/CO**



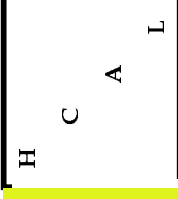
DIM (II)

H
C
A
L

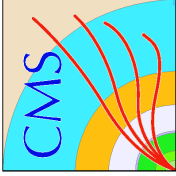




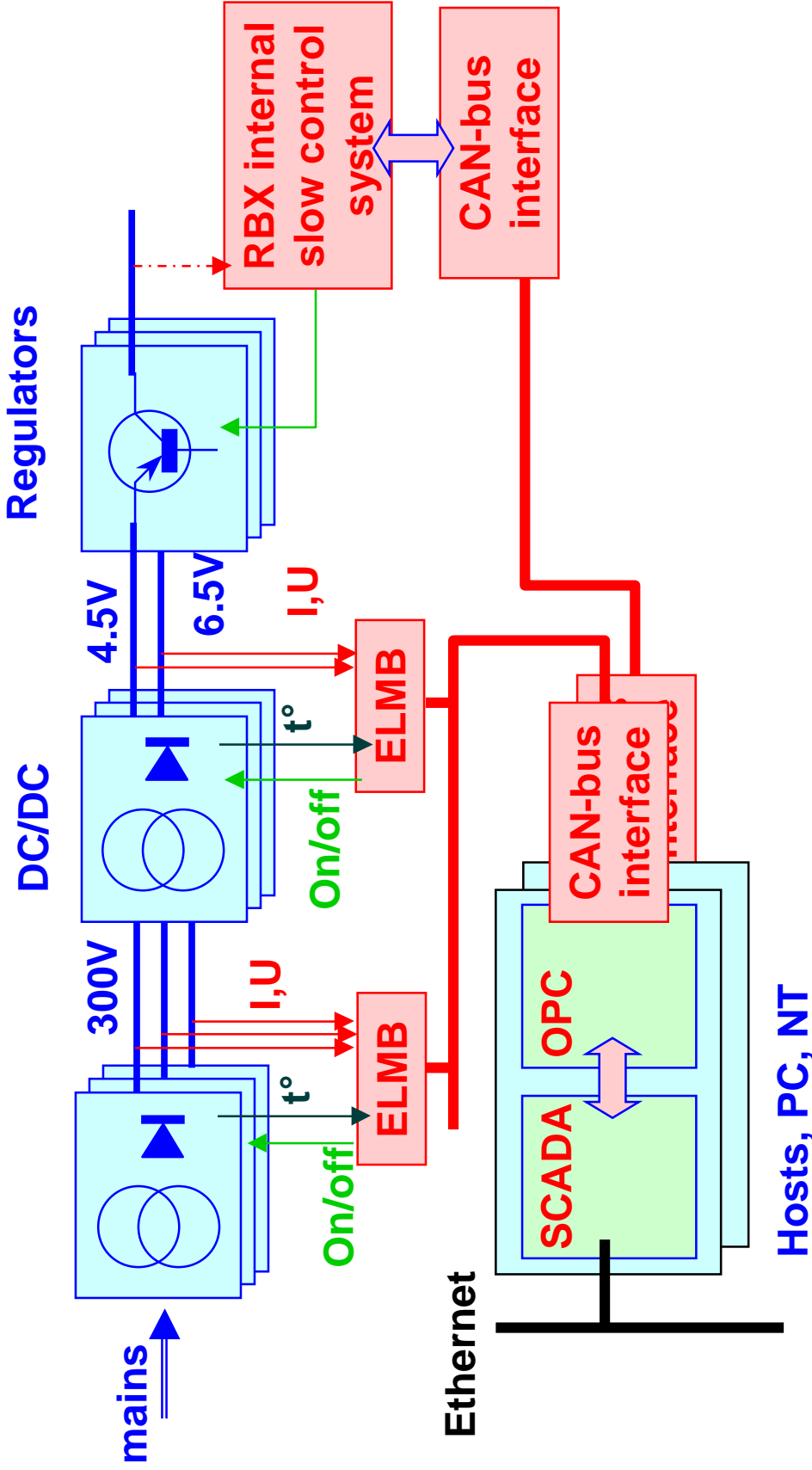
Plans (CERN)

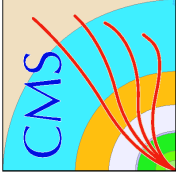


- **Low Voltage**
- **High voltage**
- **Radioactive source calibration and control**
- **Laser**
- **LED**



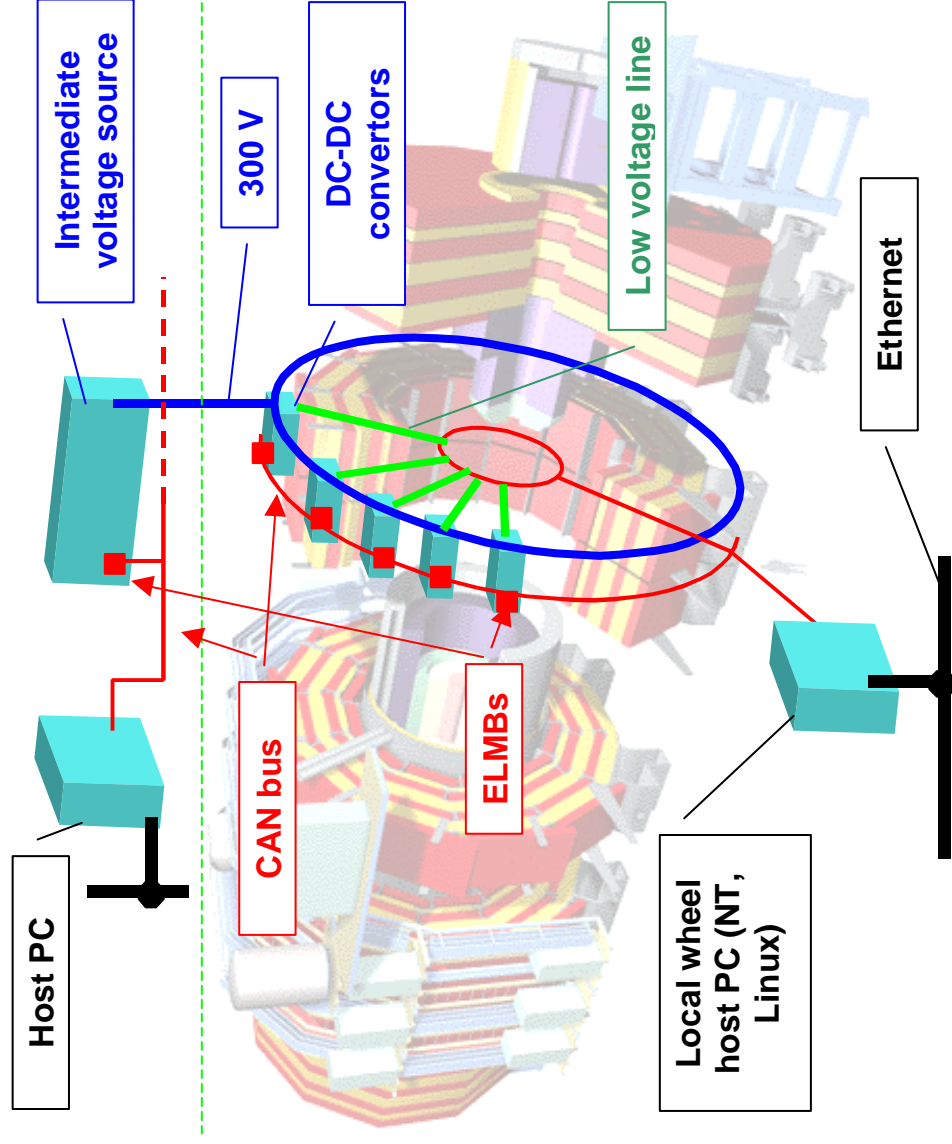
Low Voltage control system (II)

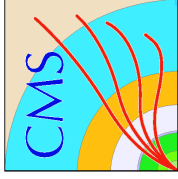




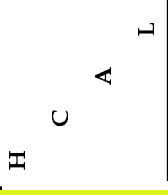
Low Voltage control system

H
C
A
L





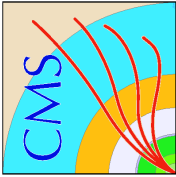
HCAL tree (URGENT)



To start development of the HCAL DCS we need:

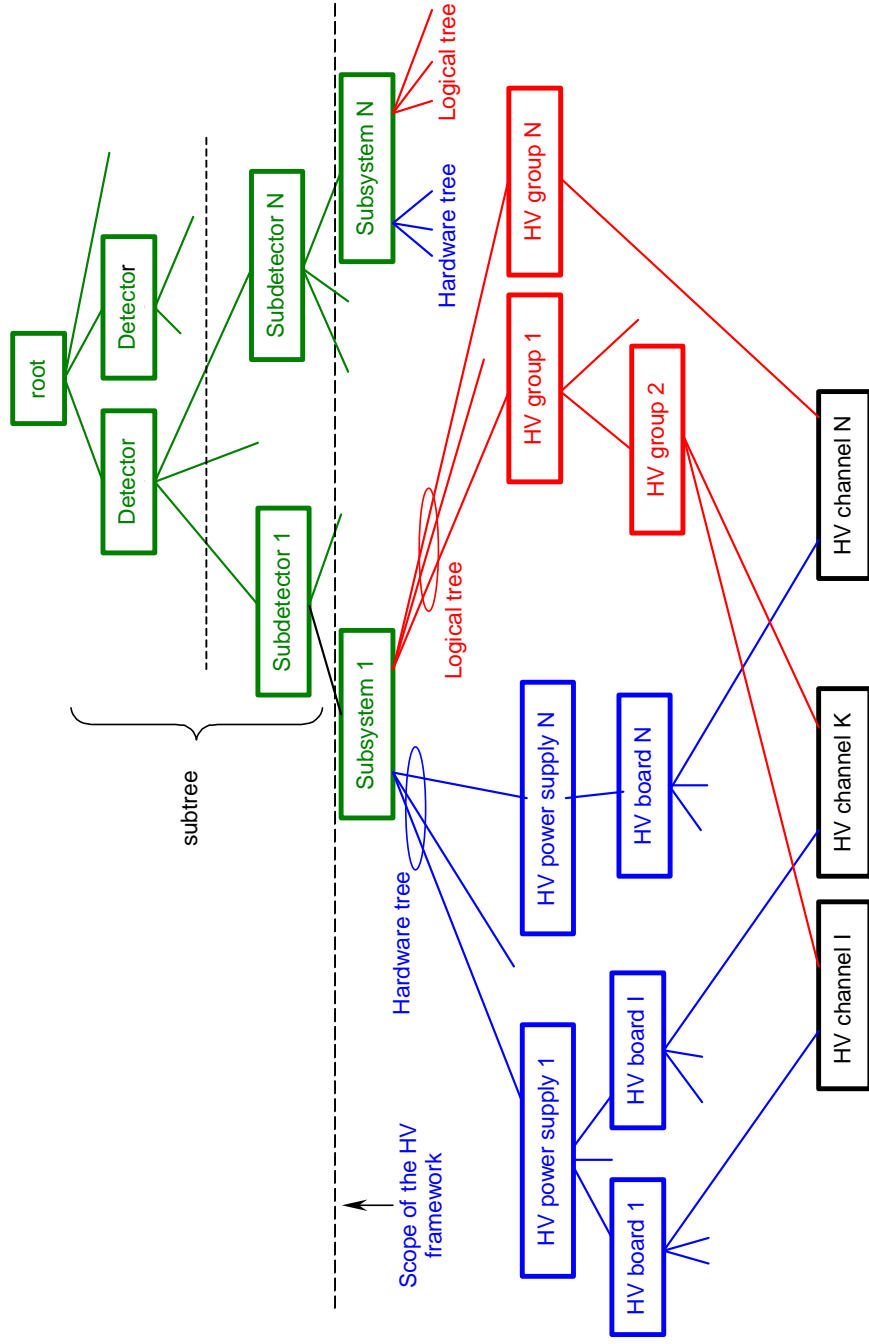
- **To draw a tree of subsystem,**
- **To define the partitioning rules**

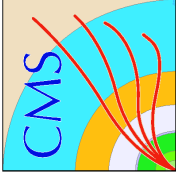
Full list of the equipment is needed



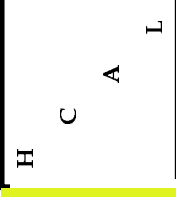
HV power supply tree

H
C
A
L

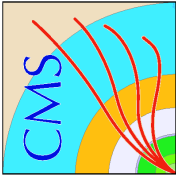




Recommendations

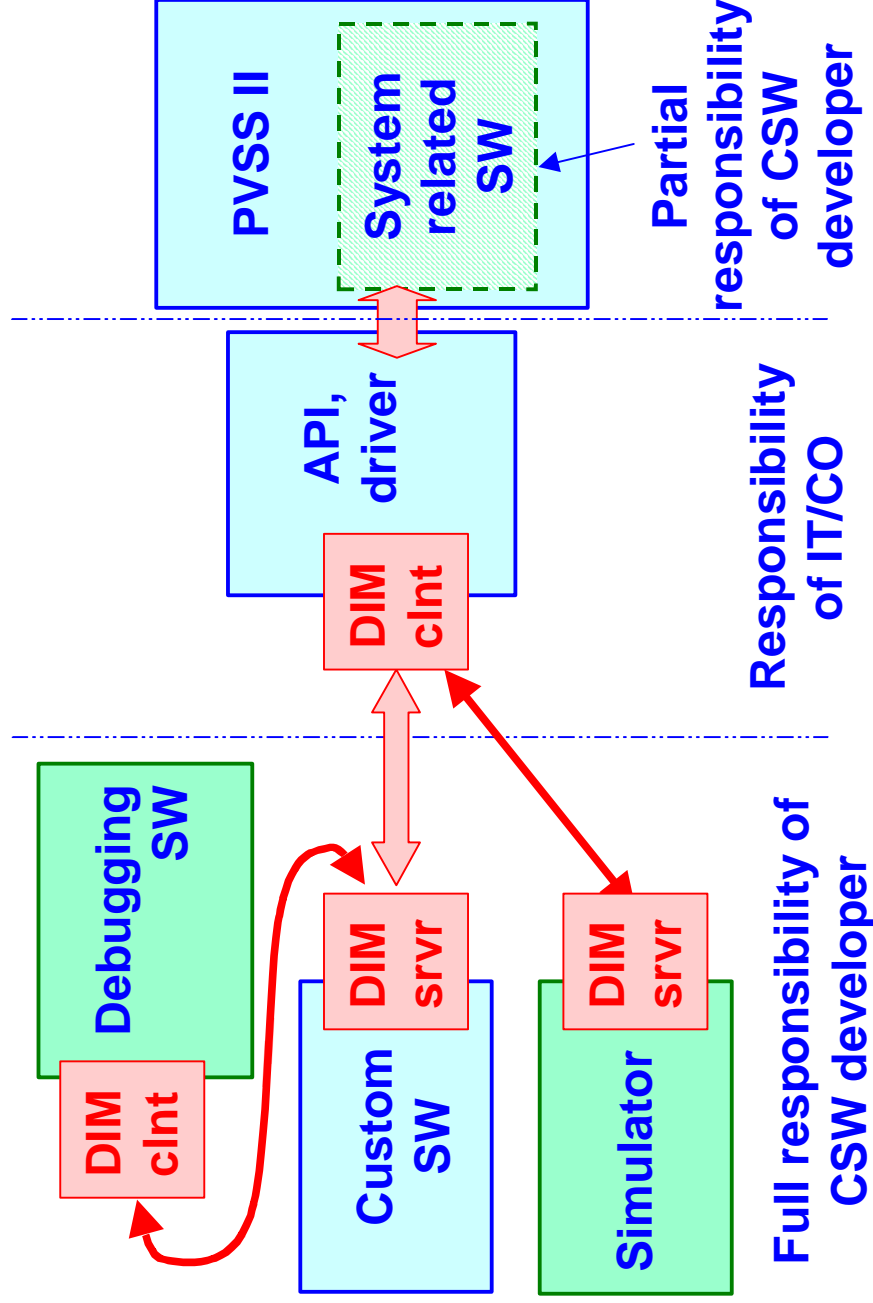


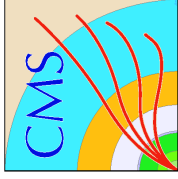
- 1. Use DIM**
- 2. The responsibility of CSW developer:**
 - to provide interface to PVSS (DIM is better)**
 - To provide debugging tools**



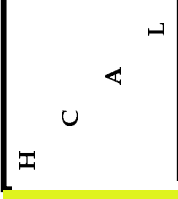
Recommendations (II)

H
C
A
L

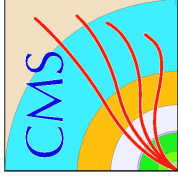




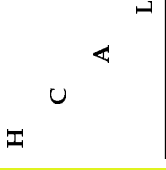
Software tools



- **It is evident that the HCAL community is short of software manpower.**
- **It is obvious that the programming language should support OOP.**
- **It is very desirable that the selected programming tool follow the "visual programming" paradigm as the most advanced and most productive method.**
- **It is extremely desirable that the selected programming tool works on both WinNT and Linux platforms in a similar way.**



Kylix (Delphi for Linux)



Very recently, Borland introduced a new product, KYLIX, an OO Pascal based RAD tool running under Linux, in fact, the only RAD tool for Linux to date. Such tools are optimized for productivity and utilize the object methodology intensively.

As an example, under Delphi/Kylix, DIM could be implemented as a standard visual component, allowing a client or server to be built using drag and drop!