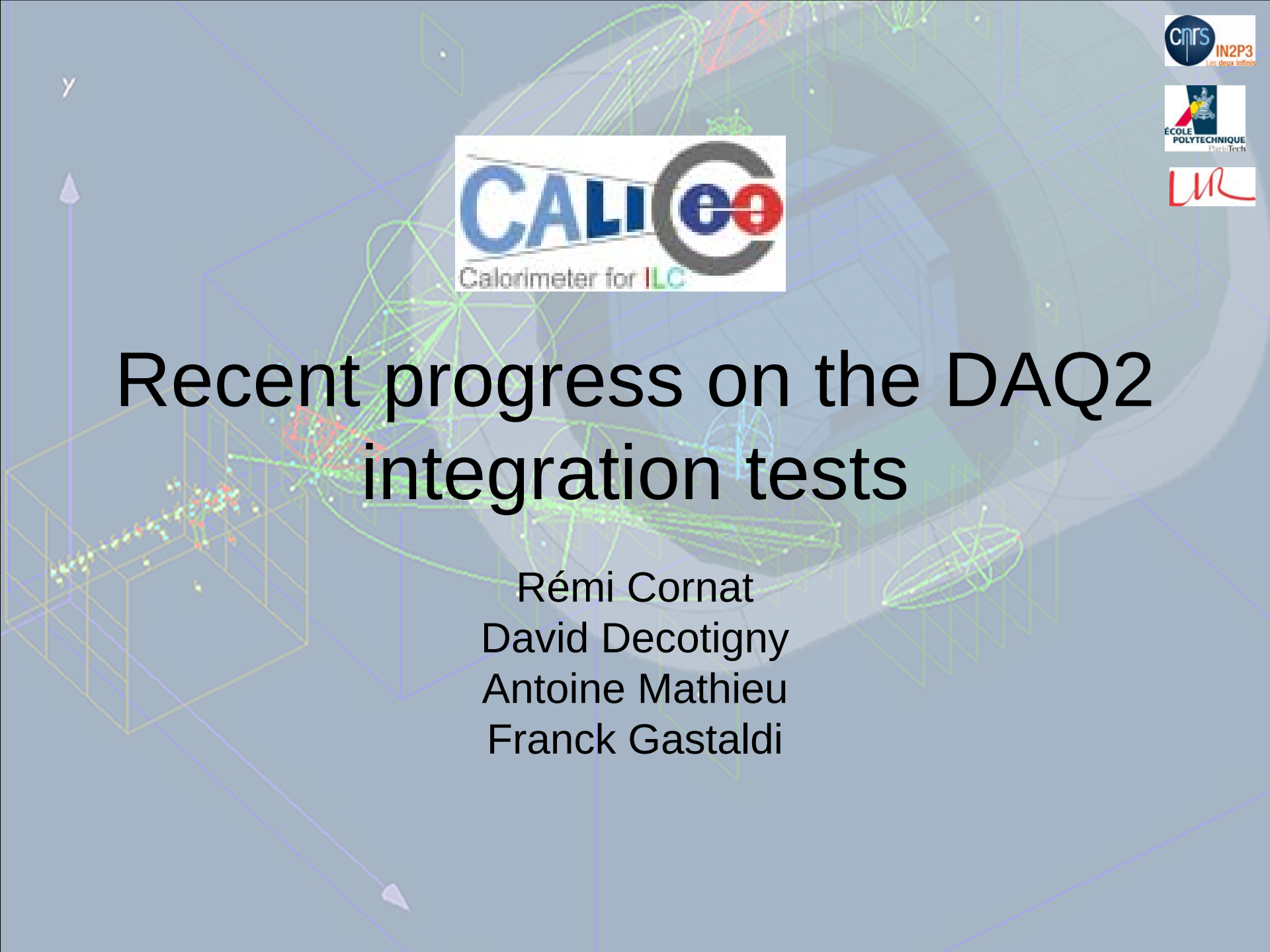


Recent progress on the DAQ2 integration tests

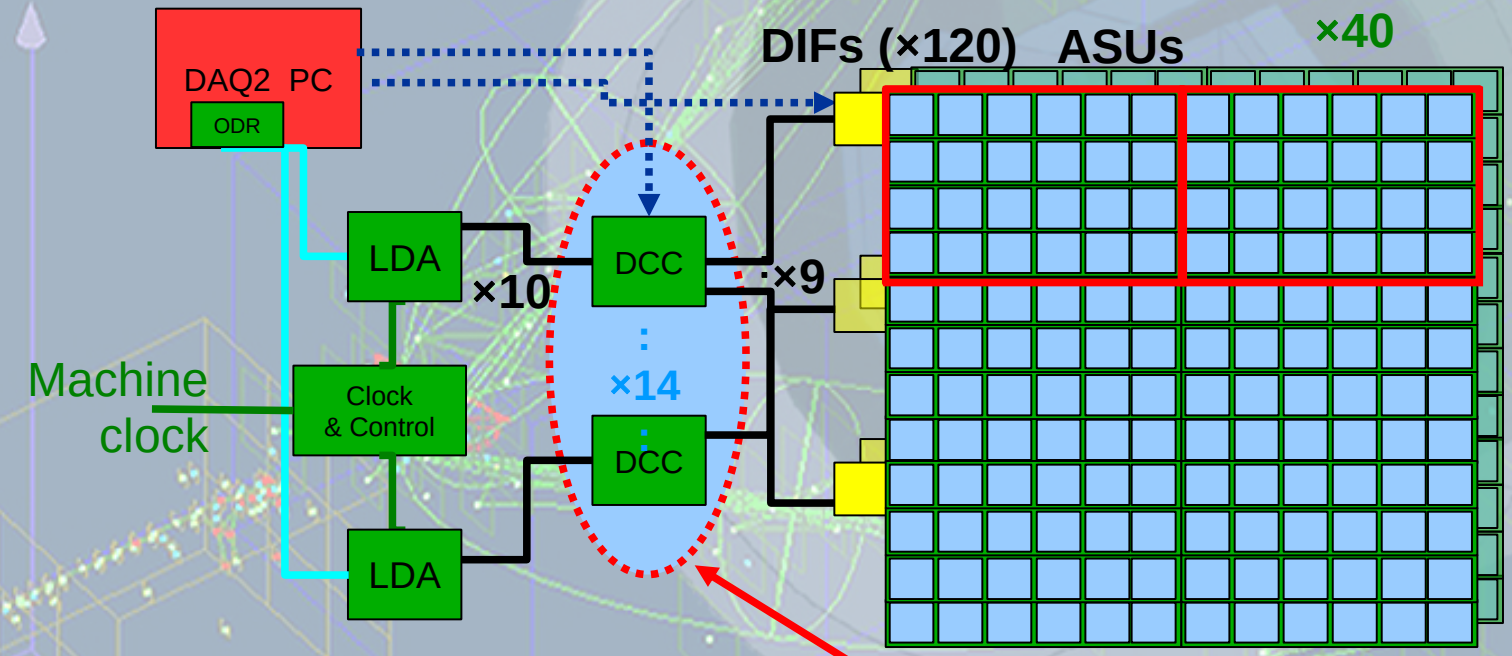
Rémi Cornat
David Decotigny
Antoine Mathieu
Franck Gastaldi

Outlines

- Hardware/Firmware part (Franck)
  - DAQ overview
  - DCC status
  - Boards on DAQ Chain
  - To do
  - Planning

  Status SW/HW interactions
- Software part (David)
  - Status and available features
  - Tests and performance results
  - Ongoing efforts and perspectives

DAQ overview



Digital (Config, Control, Data, trigger, spill) over HDMI

Clock & Sync over HDMI

Optique GigE

Debug USB

DCC being developped at LLR

Without DCC 120 DIF → 12 LDA → 4 ODR

With DCC 120 DIF → 14 DCC → 2 LDA → 1 ODR

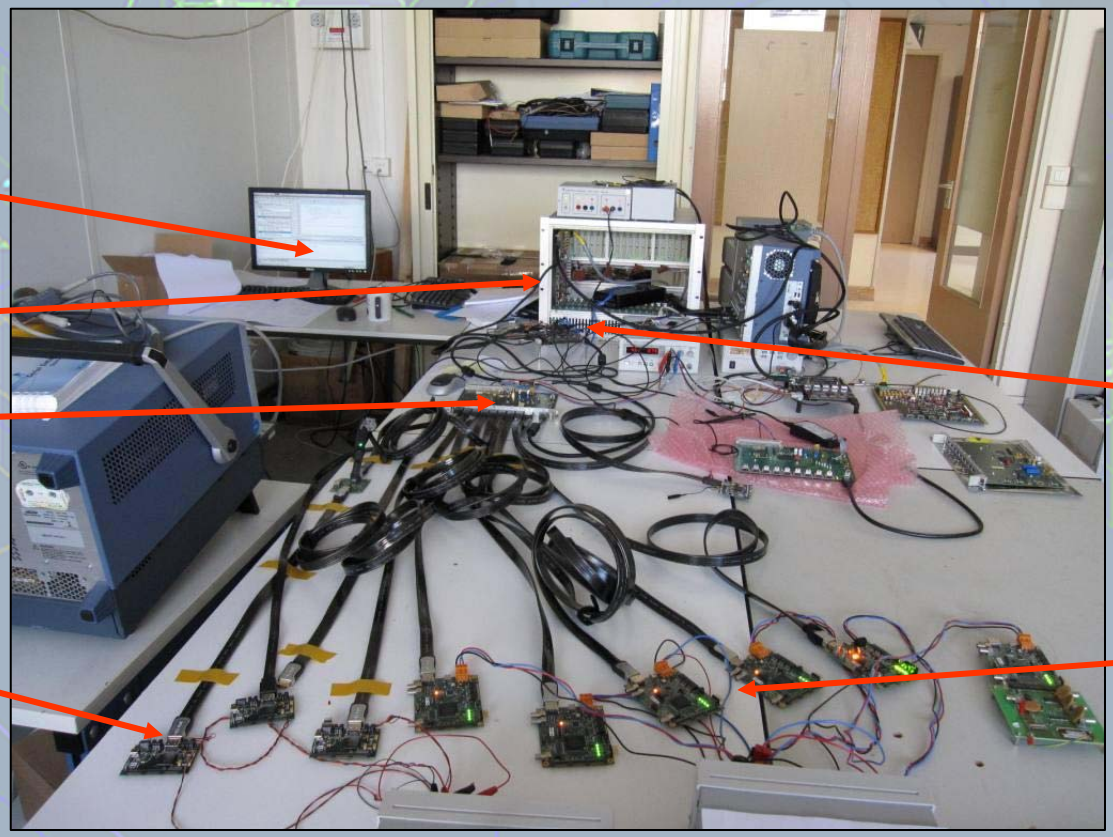
DCC at LLR

- We have achieved our objectives
  - Concentration of 9 DIFs
  - Most transparent on DAQ chain
  - Firmware functionnal for the BTCMD[1] & FCMD[2] distribution and read-out data
- DCC are in production step
  - 2 preproduction boards were delivered last Thursday (first tests are conclusive)
  - 18 boards will be launch in prodution after a group of tests (4 weeks needed for the production)
    - See David talk for the type of tests
- DCC Electrical specifications
  - 1 HDMI input : LDA connection
  - 9 HDMI outputs : DIFs connections
  - 1 USB connection (Read/Write for debug)
  - *Option* 1: 1 external memory (SRAM) of 18Mb (FW not implemented)
  - *Option* 2 : 2 external connections HE10 (26 pins)
  - Link data rate : 50 Mb/s
  - 5 LVDS signals on each HDMI

[1] BTCMD : Bloc Transfert Command , [2] FCMD : Fast Command

DAQ at LLR

PC ⇔LDA ⇔DCC⇔DIFs ⇔ (soon : ASU)

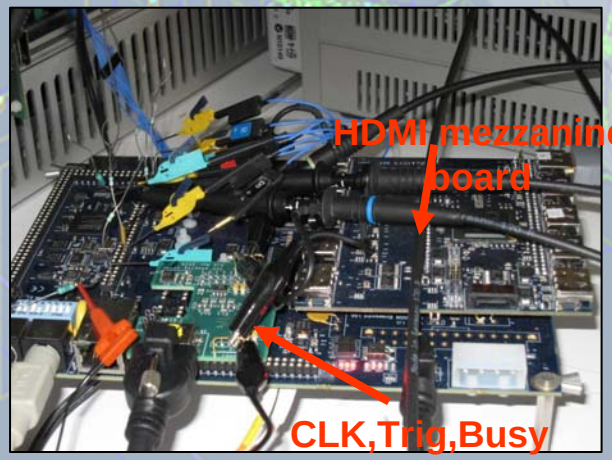


Boards on DAQ Chain

- DCC : previous slides → under control (LLR's responsability)
- ODR (UK): since july, replaced by an Ethernet commercial board (optical link connexion)
- CCC (UK): Firmware seems to be fully functionnal, maybe recquire an update for busy and trigger signals (next slide)
- LDA (UK → LLR) : see slide 7
- DIF (UK,LLR, LAPP,DESY): see slide 8

LDA on DAQ chain

- Matthew, Matt and Bart came in July at LLR to work on LDA firmware.
- What have we done during this day ?
  - Connect the clock,trigger,busy mezzanine board on LDA
    - Signals come from the CCC. Clock is correctly distributed but trigger and busy are not operationnal
  - Update the firmware for that LDA works with 10 DIFs or DCC. We have seen that channels didn't work together, probably due to a problem on multiplexer vhdl code
- With our setup tests , we have found 2 others working mistakes.
  - Back to back packet for a BTCMD (needed a delay between 2 packets)
  - FCMD sent during a BTCMD transfert (sometimes we lose 2 bytes of BTCMD packet)



LDA under debug investigation

- After a lot of investigation on LDA we have solved the problem on back to back packet.
- Problem on FCMD and multiplexer are under control and will be resolved in few weeks after the end of DCC preproduction tests board.
- UK people work on trigger and busy signals distribution.

DIF on DAQ chain

- Guillaume came at LLR at the end of June to update his DIF firmware.
- The results of tests are :
  - We have succeeded to send FCMD and BTCMD on DIF via the LDA and DCC.
  - We have succeeded to receive on PC a data packet sent by the DIF.

To do list (boards)

- DCC : Make the tests on preproduction board and launch the production
- LDA : We have enough knowledge to work on the LDA firmware. But we ask that UK team help us in case of need.
- DIF : Guillaume will come to the LLR to update and test its DIF firmware.

To do list (equipment)

- Select a mechanical frame for the CCC, DCC & LDA
- Select the power supply bloc
- Choose the HDMI cable (DCC → DIF, DCC → LDA) and required length between the mechanical frame and the DIFs.

Planning (estimation for HW/FW)

- October 2010 :
  - Final DCC production
  - Reach a functional LDA firmware
    - To resolve Mux,Trig,busy & corner effect
- November 2010 → end 2010:
  - DCC tests
  - Estimation : ASU integration

- → See David's perspective

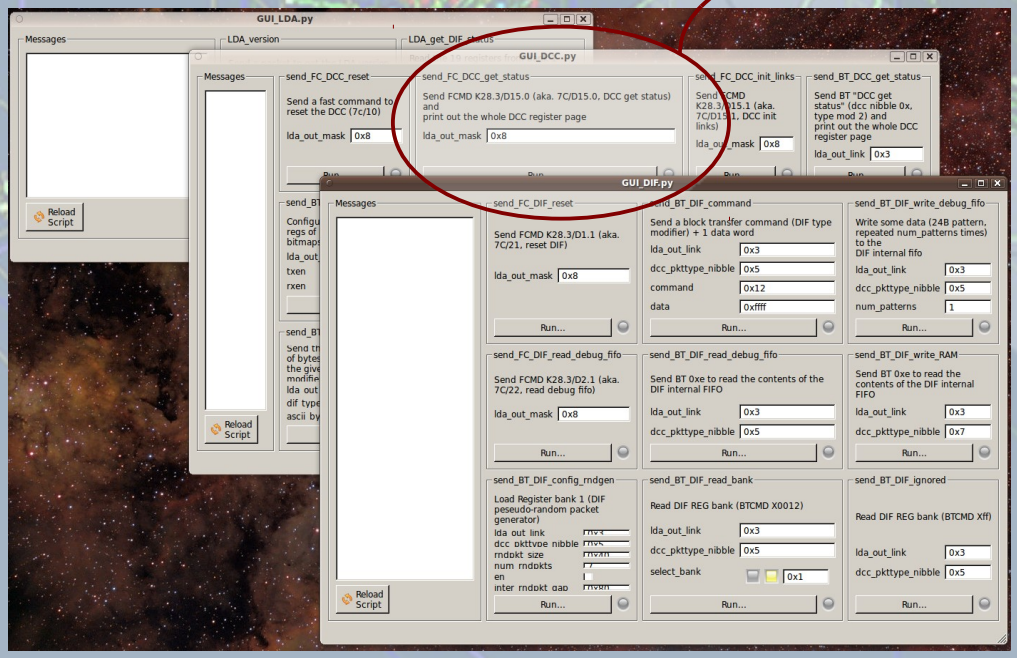
Remark : firmware can be found at :
https://svn.in2p3.fr/calice/hardware/trunk/

Status
SW/HW interactions: connectivity

- Low-level connectivity tests
  - Width LDA, over both electrical + optical ethernet PC ↔ cable
    - Register accesses
  - With DCC
    - LDA<->DCC (x1)<->DIF (x1..9) block-transfer/fast-command/read-out
    - Obsolete (was for debug): USB Subsystem low-level access (register read-write)
  - With DIF
    - Via LDA (single DIF/DCC) + optionally through DCC (1 to 9 DIFs)
      - Block-transfer both directions
        - » PC → DIF: integrity OK
        - » DIF → PC: integrity OK
      - Fast command
        - » decoded by DIF
    - Obsolete (was for debug): USB Subsystem low-level tests (register read-write)

Status
Available Software

- Interactive hardware test software (GUI)
  - Each HW test easily scriptable: simple user-friendly python API: each function defined ↔ 1 graphical pane with “Run” button
  - Intensively used by Franck/Remi
  - Available to anyone working with USB/RS/Ethernet devices




```
File  Edit  Options  Buffers  Tools  Python  Help

def send_FC_DCC_get_status(INT0x_lda_out_mask = 0x8):
    """Send FCMD K28.3/D15.0 (aka. 7C/D15.0, DCC get status) and
    print out the whole DCC register page"""
    comma  = commons.encode_8b10b_kd(28, 3)
    data   = commons.encode_8b10b_kd(15, 0)
    ans    = LDA.do_lda_send_fastcmd(INT0x_lda_out_mask, comma, data)
    calicediag.GUI.set_statusbar_message("Get_Status FCMD sent")

    return _unpack_DCC_get_status_page(ans[16:]) is not False

--:---  DCC.py      47% (189,0)   SVN-1428  (Python)----------
calicediag.register_action(DCC.send_FC_DCC_reset)
calicediag.register_action(DCC.send_FC_DCC_get_status)
calicediag.register_action(DCC.send_FC_DCC_init_links)

calicediag.register_action(DCC.send_BT_DCC_get_status)
calicediag.register_action(DCC.send_BT_DCC_config_tx_rx)
calicediag.register_action(DCC.send_BT_DCC_start_RTT)
calicediag.register_action(DCC.send_BT_DCC_stop_RTT)
calicediag.register_action(DCC.send_BT_DCC_relock_DCM)
calicediag.register_action(DCC.send_BT_DCC_register_blob)

--:---  GUI_DCC.py  Bot (7,0)     SVN-1428  (Python)----------
```

https://svn.in2p3.fr/calice/online-sw/trunk/pyserdiag/

Status
Available Software (ctd.)

- Low-level LDA+USB/DCC/DIF packet dissection/manipulation API
  - Follows LDA + future DIF specs (DIF task force)
  - Python & C/C++
  - https://svn.in2p3.fr/calice/online-sw/trunk/daq/calice_packets/
- Low-level LDA/DCC/DIF stress tests (C++)
  - More on this in next slides
- Real-world read-out simulation (1 PC to simulate DIFs+LDA, ethernet, 1 PC read-out packet reassembly/storage), based on SystemC library
  - https://svn.in2p3.fr/calice/online-sw/trunk/daq/

End-to-end tests: Pseudo-Random Generator

- Setup:
  - 9x DIF → 1x DCC → 1x LDA → PC
  - 4 DIFs generate pseudo random data
  - PC receives 4 continuous & interleaved flows of data from the 4 DIFs, checks that the sequence is correct for each DIF
  - Tested both electrical and ethernet LDA ↔ PC connection
- Results:
  - Direction DIF → LDA OK, even in multi-DIF thanks to DCC
  - Maximum DCC → LDA link occupancy (40Mbps) OK
  - Up to 5.6 TB transferred (2 weeks), no error

End-to-End tests (ctd.): FIFO Write/Read

- Setup:
  - PC ↔ 1x LDA ↔ 1x DCC ↔ 1x DIF
  - PC fills DIF FIFO with data, reads FIFO back and compares, then starts over
  - Tested both fast-commands and block transfer “read” requests
  - LDA back-to-back Ethernet packet reception problems → inserted >= 1ms pause between PC → LDA packets
- Results
  - Direction LDA → DIF requires precaution, but reliable
  - BER < 8.10^{-12} bit^{-1}
  - Mean Bandwidth (overnight): 3Mbps
    - LOW due to 1ms pause between packets & OS latency

End-to-End tests (ctd.)

- Summary:
  - ✓ Low-level HDMI links stable over long periods of time
  - ✓ multi-DIF → 1x DCC → LDA fully working: reliable and reaches theoretical maximum performance
  - ✓ LDA ↔ 1x DCC ↔ 1x DIF reliable, with LDA → DIF limitations (fixed but not re-tested: see next slides)
  - ? Multi-DIF/Multi-DCC → LDA operation: next slides

Ongoing efforts

- DCC development support: done (proto + prod)
- LDA debug: in progress
  - A few points we are working on:
    - Fix back-to-back PC→LDA packets bug: Fixed for BT
      - Tests need to be re-run
      - On-going for fast-commands
    - Multi-DIF operation: soon (based on Franck's round-robin ?)
    - Not a SW issue: buffer overflow on non-connected sinks
  - Foreseen end-to-end tests:
    - Higher throughput FIFO test (no pause between packets), multi-DIFs (multi-threaded)
  - ➔ We are gaining confidence the chain will be OK soon !

Perspectives

- Integration with Lyon's setup
  - Control of Guillaume's SDHCAL DIF Firmware
  - Integration with Christophe/Laurent's XDaq DAQ system for the SDHCAL m³ prototype