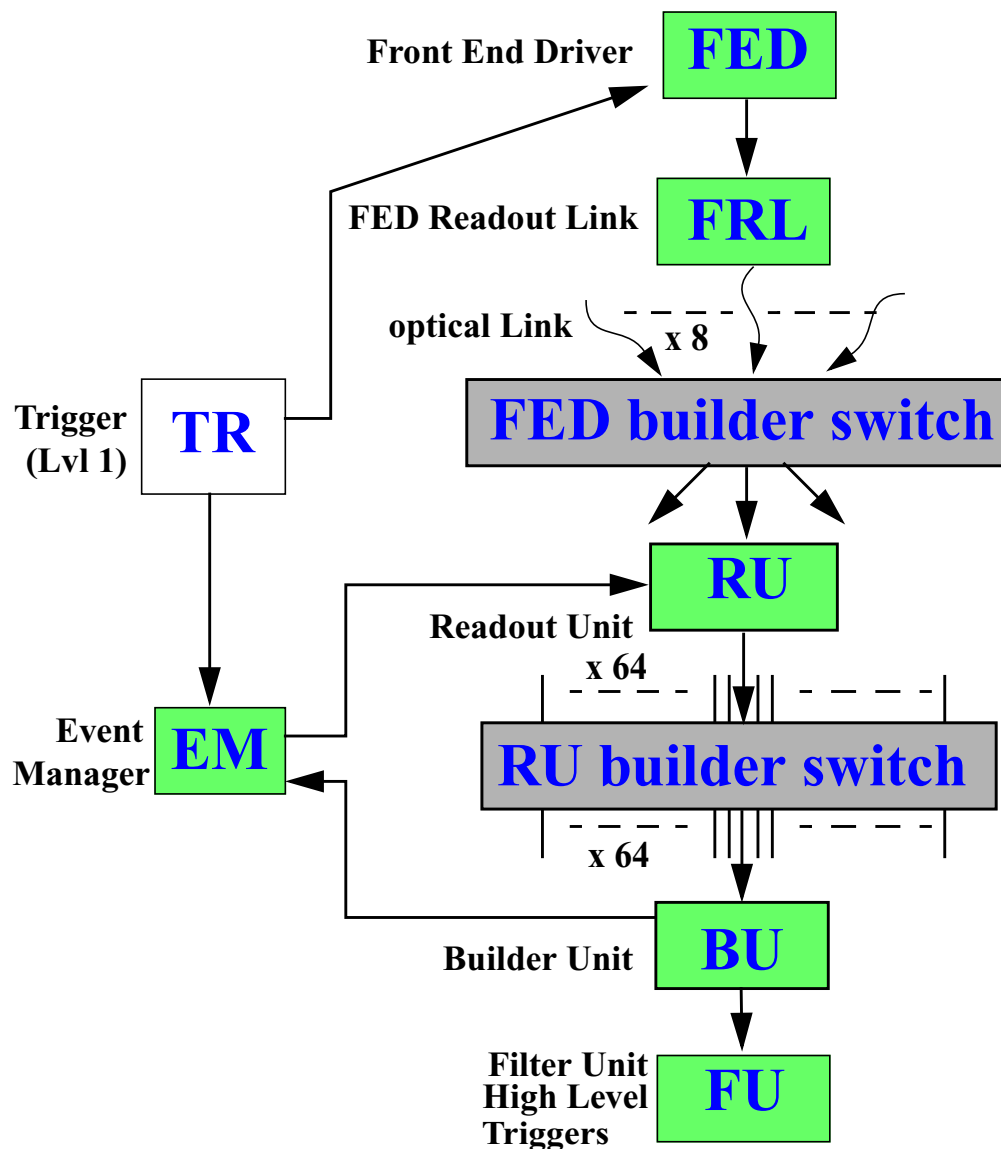


DAQ Column

- What do we want to have in the end?
- What do we want next
 - What do we have (can we reuse)
 - What do we have to provide
- Other testbenches (related to

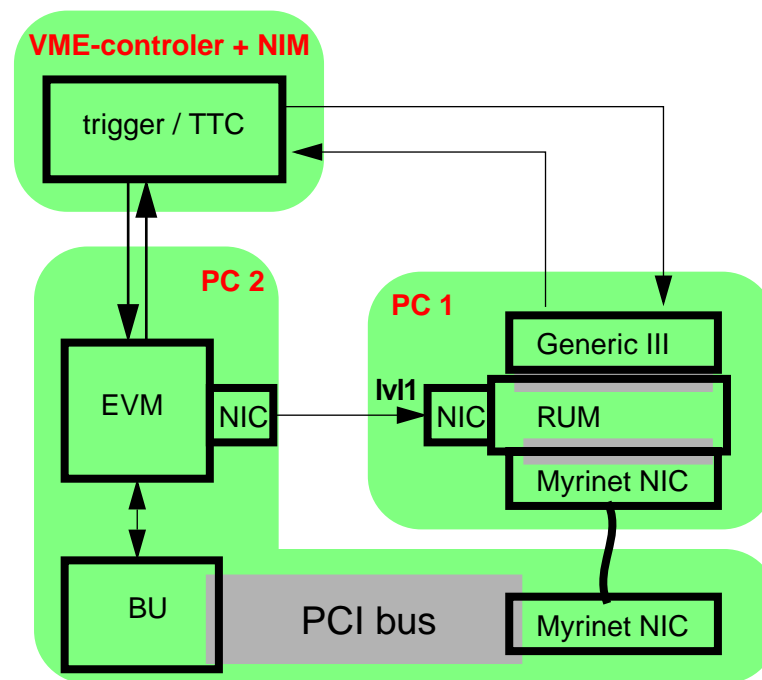
What do we want to build

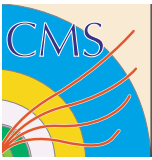
- One of each component of the DAQ system
 - test out different implementation choices
 - measure performance under possibly realistic circumstances
 - find the right choice for the final implementation



What is the next step ?

- Aim is to measure data throughput through the Readout Unit implemented with a PC.
 - Data input
 - Data output (Myrinet)
 - Builder Unit
 - Event Manager (RM and BM)
 - Trigger logic (TTC)
 - EVM and BU could be in different PCs (need a communication then EVM-BU then: could be Myrinet over switch)

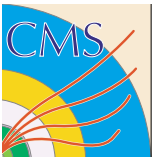




What do we already have ?

- A simple column had been built and tested
- Components of that system

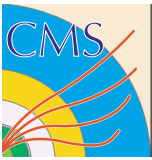
component	reusable?
Trigger Logic (NIM) with VETO input	yes
TTCvi and optical distribution	yes (use Hardware Access Library)
TTCrx hardware with readout software	yes (with software modification)
RUI card based on generic II as data input	no
RU in XDAQ	latest version
EVM in software	latest version
Control software	with modifications (STM,...)



What do we need to provide?

- Hardware components
 - Two or three PCs (2 are in place)
 - Generic III with trigger and veto in / out easy task (hardware in place)
- Software components
 - Integration of Hardware Access Library / i2o-core Akos, Christoph
 - Control software update Akos, Christoph
 - FED toolkit for data input, integration in XDAQ Eric, Johannes, Luciano
 - RU / BU modification to fake 64 sources and 64 destinations Johannes, Lucaino
 - Monitoring software for measurements Akos, Christoph
- Firmware components
 - NICs for RUO / BU connection Frans

Plan: get things running until end february; measure in march
But : depends on a lot of factors and the workload of a lot of different people



RU / BU modifications in order to get realistic traffic

- Number of messages sent (control and data) should be like in “real” system

real system

BU sends for each event 64 requests for fragments from 64 different sources to 64 different RUs

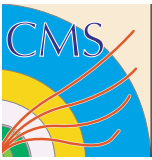
RU sends 1 fragment for each request to one of 64 possible destinations. The total fragment rate is maximal 12.5 kHz

DAQ Column

BU sends for each event 64 requests for fragments from 64 different sources to **the only** RU in the column

RU sends 1 fragment for each request to **the only** BU. It fakes the event source id according to the event source requested by the BU.

- The rate of messages is the same in both cases
 - The BU has to do the same job in the column as in the real system (eventbuilding by doing pointer gymnastics)
 - The data throughput of the RU is the same as in the real system (12.5 kHz fragement rate)
- Not realistic:
 - The order in which the messages arrive in the column is always the same and not “randomized” due to event builder traffic.



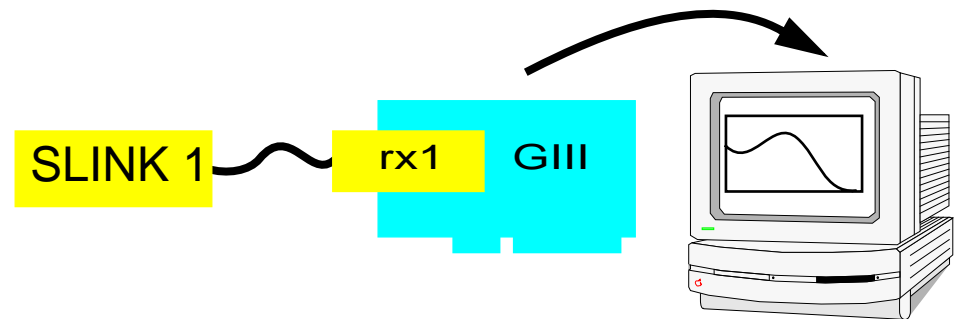
Further development of the column

- Status of this column
 - At the beginning: hardware exists, work-plan (who does what) has been made
- Extend the column towards the FED:
 - Integrate Myrinet link at the RU input
 - Connect the Input of Myrinet link to a FRL
 - Connect the BU to a filter farm
 - integrate EVM and RCN network of Fermilab
 - build a trigger processor emulator with a GIII
 - integrate the column in the Event Builder Demonstrator so that it forms one leg together with the dummy legs of the demonstrator

In parallel: other testbenches

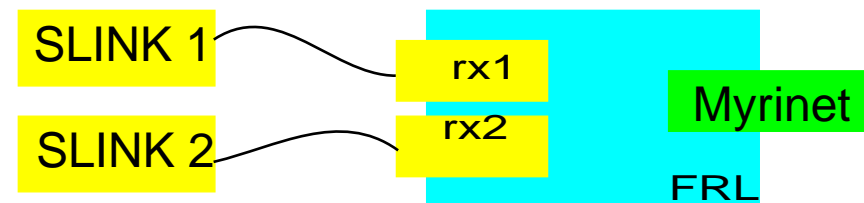
- **FED-DAQ-KIT**

- desing is done
- implementation is being debugged currently
- integration into XDAQ needs to be done



- **FRL testbench**

- Hardware has been designed, being built
- Software / Firmware is being designed



- **Demonstrator tests**

- Compare XDAQ based demonstrator with current version
- Compare two different packeting schemes:
 - packeting is done in the host
 - packeting is done by the NIC
- see talk of Frans