Thomas Ndousse, DOE ASCR Program Manager, SLAC Visit August 21st, 2012

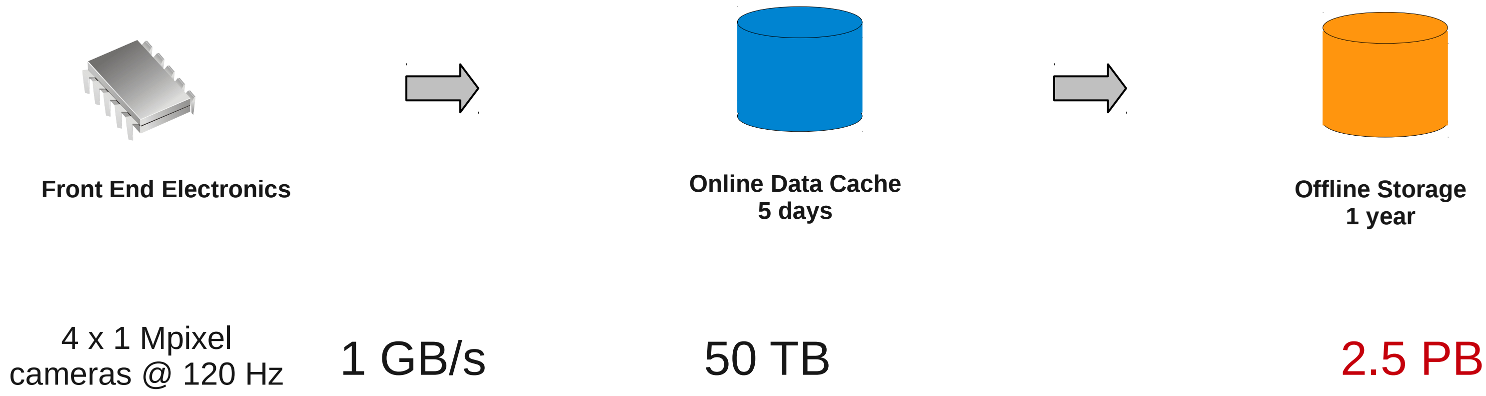
LCLS Data Collection

Amedeo Perazzo

FEL Data Systems Key Challenges

- **Ability to readout, event build and store multi GB/s data streams**
- **Allow experimenters to analyze data on-the-fly**
- **Flexibility to accommodate user supplied equipment**
- **Ability to store and analyze very large data sets**

Does LCLS have a Data Problem?

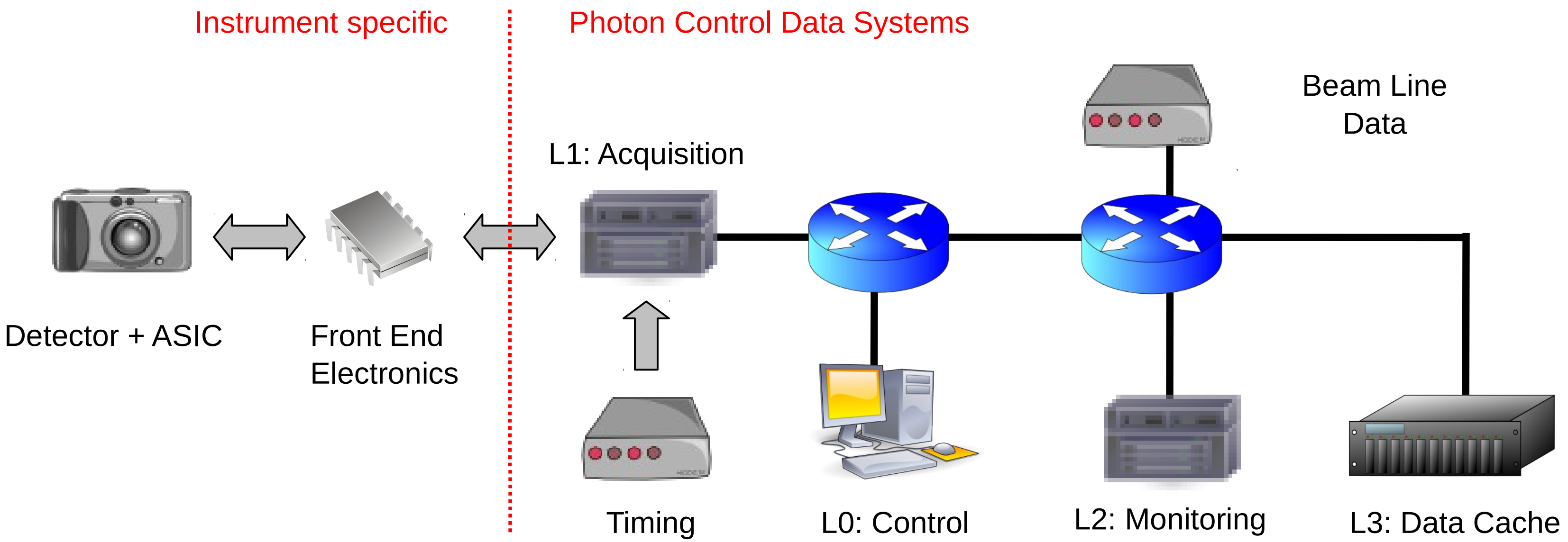


- **LCLS doesn't have a dataflow problem, yet**
  - Rate could increase x3 (operating LCLS @ 360 Hz is possible, but unlikely)
  - Bigger contributor will be introduction of larger, multi mega-pixels, detectors
- **LCLS does have a (small) storage problem**
  - Unlike dataflow, storage will increase with concurrent instrument operations
  - LCLS can afford to reduce its storage requirements by filtering and compressing the data offline

Data Rates Comparison

	Beam Rate	Trigger	Event Size	Recorded Data
LCLS	120 Hz	120 Hz	10 MB	2 PB/yr
SACLA	60 Hz	60 Hz	12 MB	
XFEL	27 kHz (10 Hz * 2700 [5MHz])	3 kHz		50 PB/yr
BaBar	238 MHz	4 kHz / 300 Hz	50 kB	1 PB/yr
ATLAS	40 MHz (20 MHz)	100 kHz / 200 Hz (65 kHz / 700 Hz)	1.5 MB (1.4 MB)	10 PB/yr (3 PB/yr)

LCLS DAQ Architecture



- **Each instrument has its own, dedicated instantiation of DAQ system**
- **Most of the customization effort goes into the readout of the instrument specific front-end electronics**
  - LCLS would greatly benefit by the standardization of the readout protocol adopted by the various detectors

Looking at Data on-the-fly: Online Monitoring

- **Online monitor framework allows users to analyze, on the fly, the quality of the data**
  - Implemented by snooping on the DAQ traffic between the readout nodes and the data cache nodes
    - Guarantees that monitoring does not impact data acquisition
- **Users can augment the existing monitoring features by dynamically plugging in their code to the core monitoring framework**

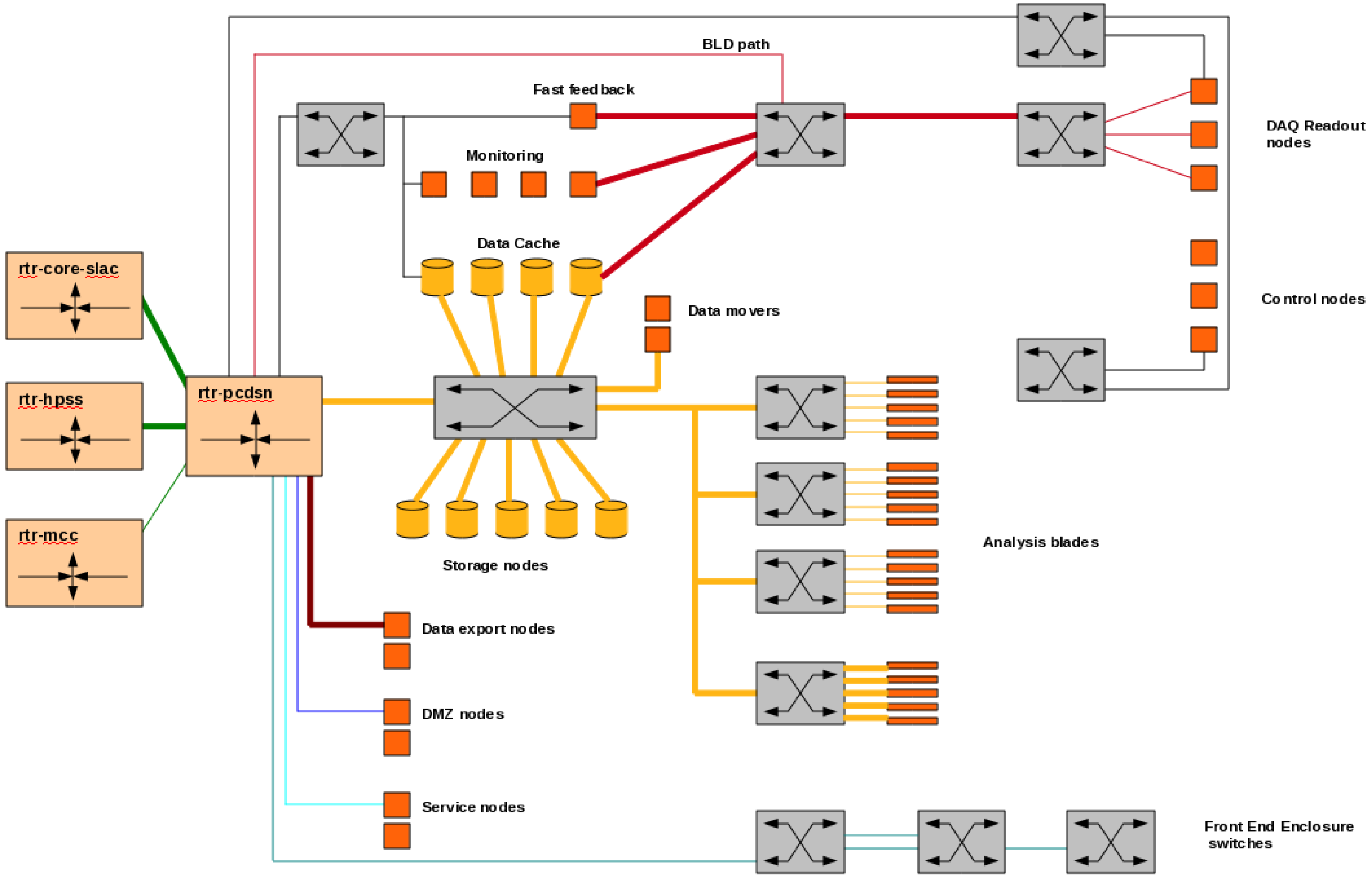
Separating Users' Activities from Data Acquisition: Online Data Cache

- **Data cache nodes:**
  - assemble the components from the different readout nodes which correspond to same pulse (event building)
  - store full event to the local RAID array
- **Data cache currently 200TB per instrument**
  - isolates DAQ system from users operations
  - allows experiments to take data even during outages of the offline system
- **Data files are copied over 10Gbps links from online cache to medium-term storage where they are made available to the users for offline analysis and for off-site transfer**

DAQ Interfaces to Other Subsystems

- **Controls: DAQ interfaces to controls in order to:**
  - store some user selected EPICS process variables together with the science data
  - control any device that can be used to perform a scan or a calibration run
- **Beam Line Data: DAQ receives small pieces of information which contain key beam measurements**
  - currently three packets per pulse:
    - e-beam parameters from accelerator, timing information from RF cavity, gas detector measurements from front-end enclosure
  - timestamped with the pulse ID and stored with the science data

LCLS Data Networks



User Data Analysis

- **Analysis system shared among the different instruments**
- **Main physical components of analysis system are:**
  - medium-term storage
  - long-term storage
  - processing farm
- **Analysis system also provides software frameworks to:**
  - copy the science data to medium and long term storage
  - translate the data into user formats (HDF5)
  - parse and analyze the data

Science Data Storage

- **Medium-term storage is disk based**
  - Current size 4 petabytes
  - Each PB has maximum aggregated throughput of 12GB/sec
  - Each client has throughput from 50 to 800 MB/s
- **Long-term storage uses tape staging system in the SLAC central computing facilities**
  - Can scale up to several petabytes
- **Science data files policies:**
  - Kept on disk for 1 year
  - Kept on tape for 10 years
  - Access to the data for each experiment granted only to members of that experiment

Data Movers

- **Experimenters allowed to transfer their data files to their home institution if they decide to do so**
  - two data mover nodes allocated for that purpose
- **Disk storage communicates with**
  - tape staging system
    - dedicated dual 10Gbps links
  - SLAC main router for off-site data transfer
    - additional dual 10Gbps links

Data Processing

- **Processing farm based on:**
  - Batch pool: 1000 cores
  - Interactive pool: 192 cores
- **Farms live in the experimental areas with fast access to the science data files in medium-term storage**
  - Batch nodes: Infiniband QDR
  - Interactive nodes: 10Gb/s Ethernet

Lesson Learned 1 or Why Vetoing Events for FEL Experiments Can Be Tricky

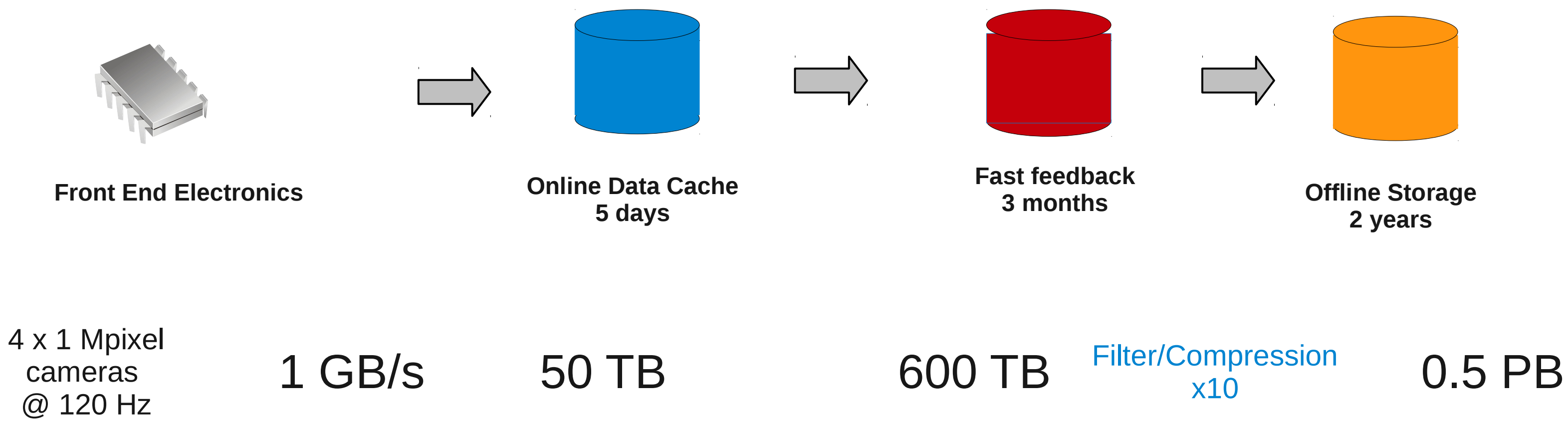
- **Very hard to implement effective trigger/veto system**
  - Not a technical/computing issue: the ability to veto events is already implemented in the system
  - Vetoing based on beam parameters not effective (most pulses are good)
  - Must be based on event features, but hard to get help from users in setting veto parameters which define event quality
    - Users themselves often don't know what these parameters or their thresholds should be
    - Users are usually very suspicious of anything which can filter data on-the-fly
- **Benefit of vetoing events based on the event features can be large**
  - For some experiments we observed factor 10-100, but this ratio will likely decrease in the future as hit rate improves (for example by improving injector technology)

Lesson Learned 2 or Why HEP Style Online-Offline is Not Enough

- **HEP style online/offline separation doesn't work**
  - The core online monitoring is not enough for many experiments
  - The skill level required to write on-the-fly analysis code is too high for most users
  - As a consequence some experiments feel they fly blind
- **Critical to provide users the ability to run offline style code for fast feedback**
  - Currently an issue for:
    - High data volume combined with low hit rate experiments: offline designed to keep up with DAQ only in average, not instantaneously; fast feedback nodes which look at subset of the data don't provide enough statistics
    - HDF5 based experiments: must wait for additional translation step

Lesson Learned 3 or When Users Can Use a Little Push

- **Plan to modify data retention policy with dual-fold goal: encourage users to filter their data and provide fast access to the data for longer period**
  - Set a quota on data kept on disk and extend the lifetime of the data on disk (1 -> 2 years)



Lesson Learned 4 or Why We Need Yet Another Software Framework

- **High fragmentation analysis tools adopted by users for data analysis**
  - psana (LCLS C++ framework), pyana (LCLS Python framework), Matlab, IDL, Igor, etc
- **Strong need of high performance, open source framework**
  - HEP community attempted something similar with ROOT, but was not fully successful
- **Should provide**
  - Way to make core objects and user data persistent (and retrieve)
  - High quality and powerful plotting, histogram, fitting tools
  - Both scripting and compiled languages
  - Algorithms needed by the photon science community

Conclusions

- **LCLS has currently manageable data problem**
  - Things will get more interesting with the planned future 16Mpixel detectors
- **Introduction disk-based fast storage layer between online and offline critical for LCLS**
- **Strong need of high performance, open source, software ecosystem for data analysis at FEL facilities**
- **Standardization of detector readout protocol would greatly benefit both facilities and detector development efforts**
  - There are many detector readout protocols available (eg, UDP, PGP, camera link), no real need to introduce new ones
  - Standardization of the protocol messages would also be extremely helpful (albeit ambitious)