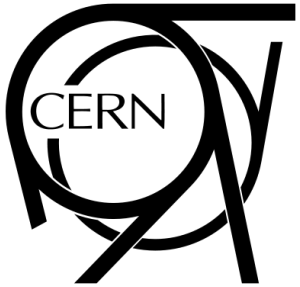
Online Software



InfiniBand enabled XDAQ clusters

CMS DAQ Upgrade Studies

17th November 2011 – CERN - SuperB ETD Meeting

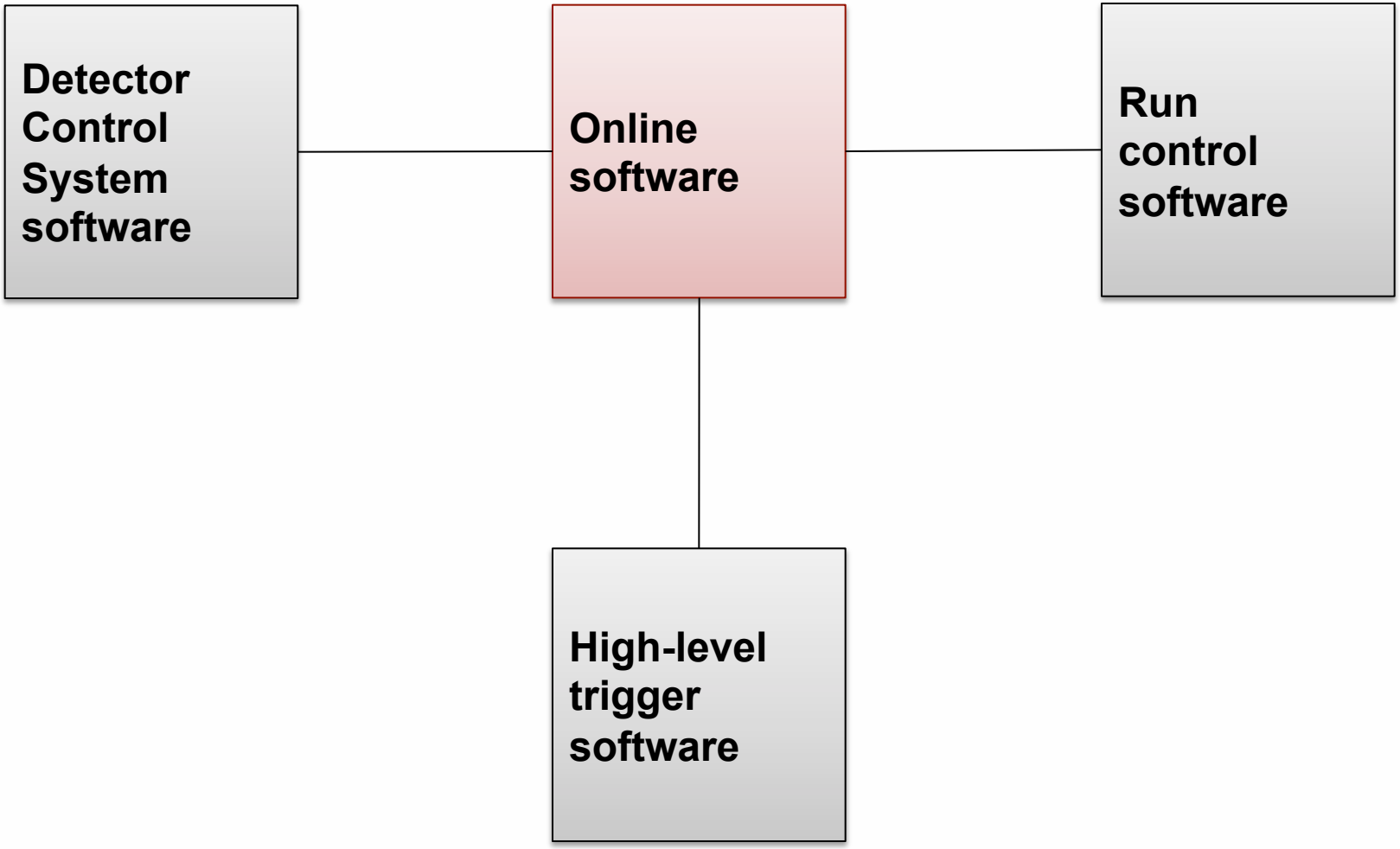
Luciano Orsini, Andrea Petrucci – CERN (PH/CMD)

Topics

- XDAQ (CMS on-line software)
  - Scope
  - Motivation and Requirements
  - Architecture
  - Software Product Line
  - Outlook
  - Configuration Management
- InfiniBand enabled XDAQ clusters
  - InfiniBand software architecture
  - Protocol stack overview
  - XDAQ peer transport architecture overview
  - Early adopters
  - The XDAQ uDAPL peer transport
  - Testbed environment for I/O benchmarks
  - Preliminary measurements
- Conclusion

XDAQ (CMS on-line software)

Context Diagram



Motivation

- CMS consists of a set of sub-projects
  - Similar to a coordinated set of small experiments
  - Many scenarios: central DAQ, subdetector DAQ, testbeams,
- Geographically dispersed participants
- Autonomous developments
- High personnel turnover
- High performance requirements
- Long lifetime and need to survive technology generations
- Similar tasks to be performed in each sub-detector

Functional Requirements (TDR)

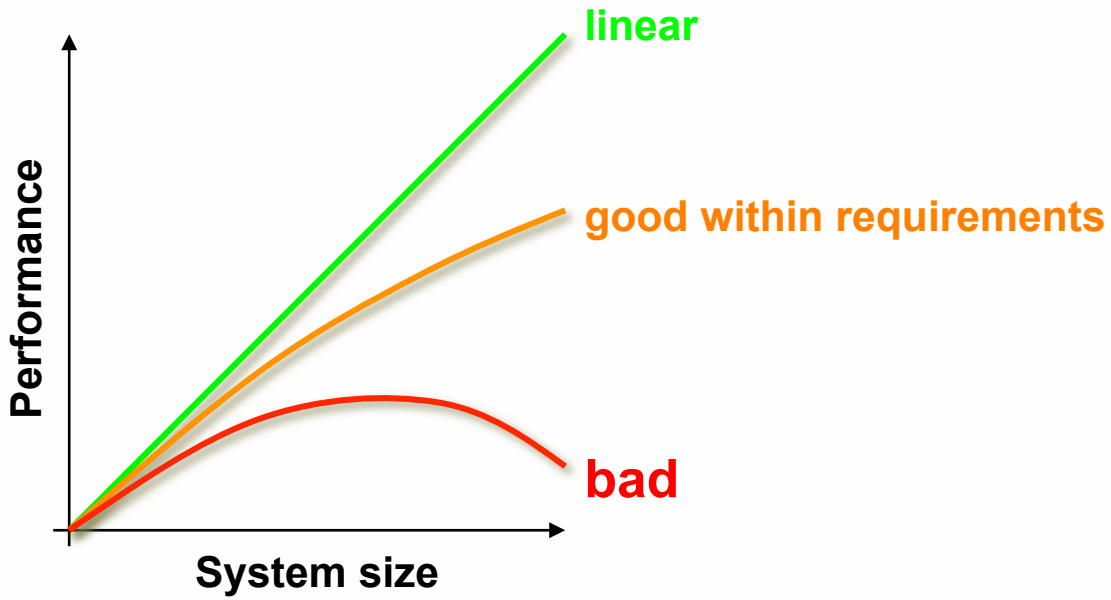
- **Communication and Interoperability**
  - Transparent use of communication protocols
  - Possibility to add new protocols
  - Concurrent use of multiple protocols
- **Device Access**
  - Access to custom devices
  - Hardware abstraction layer
- **Configuration, control and monitoring of applications**
  - Inspect and modify simple/complex parameters
  - Allow coordination of application components
  - Record structured information
    - Uniform logging, error reporting, monitoring
    - Interface to persistent data stores

Non-Functional Requirements (TDR)

- **Maintainability and Portability**
  - Portability across operating system and hardware platforms
  - Add new electronics without functional changes in user software
  - Application code shall be invariant with respect to the physical location and the network
  - Encourage working with re-usable building blocks

Scalability

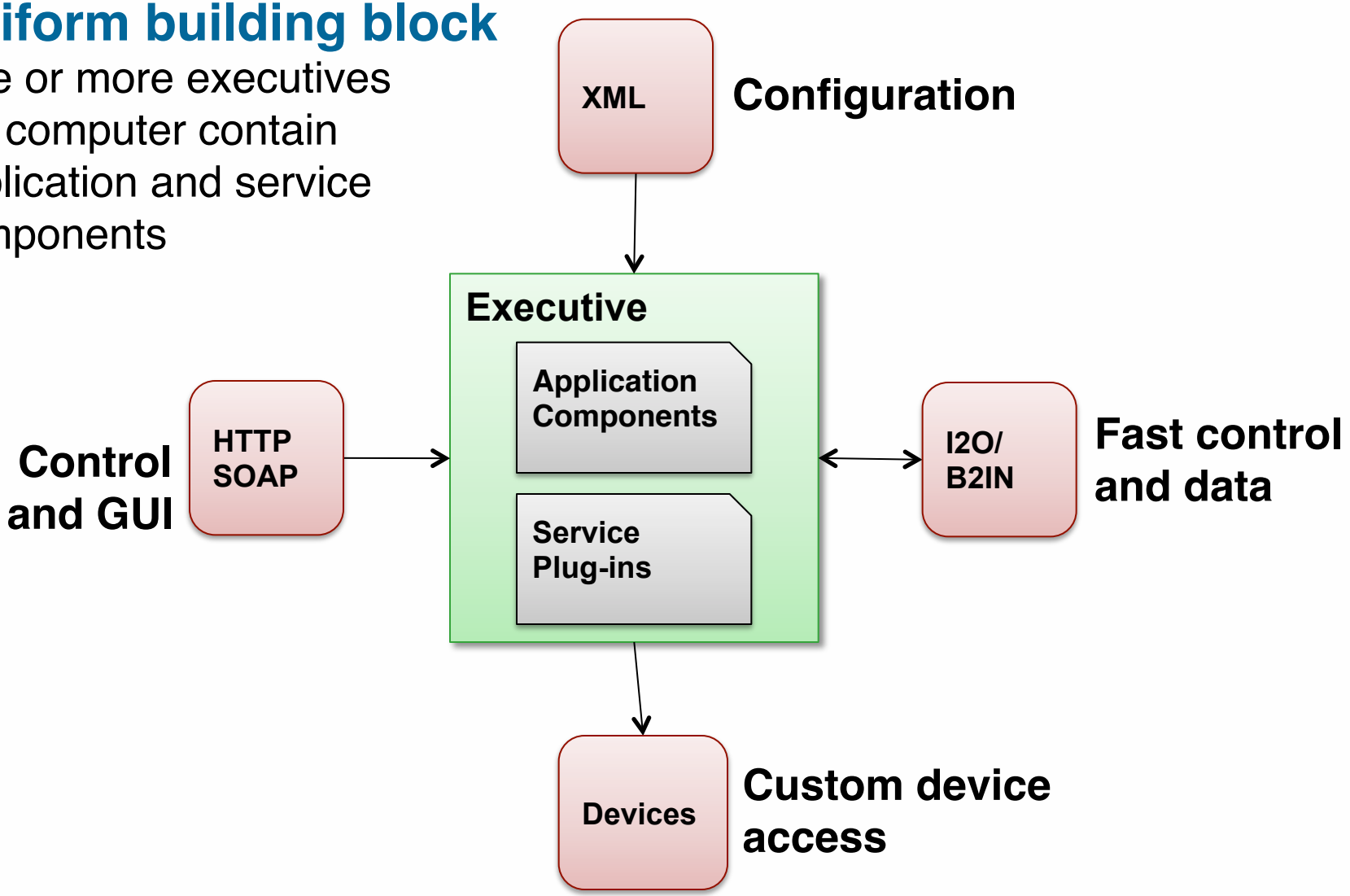
- **Scalability**
  - Operate within requirements if size or volumes change
  - Take advantage of additional resource availability
  - Overhead introduced by the software environment must be constant for each transmission operation and small with respect to the underlying communication hardware in order not to introduce unpredictable behavior



Architecture Foundation

Uniform building block

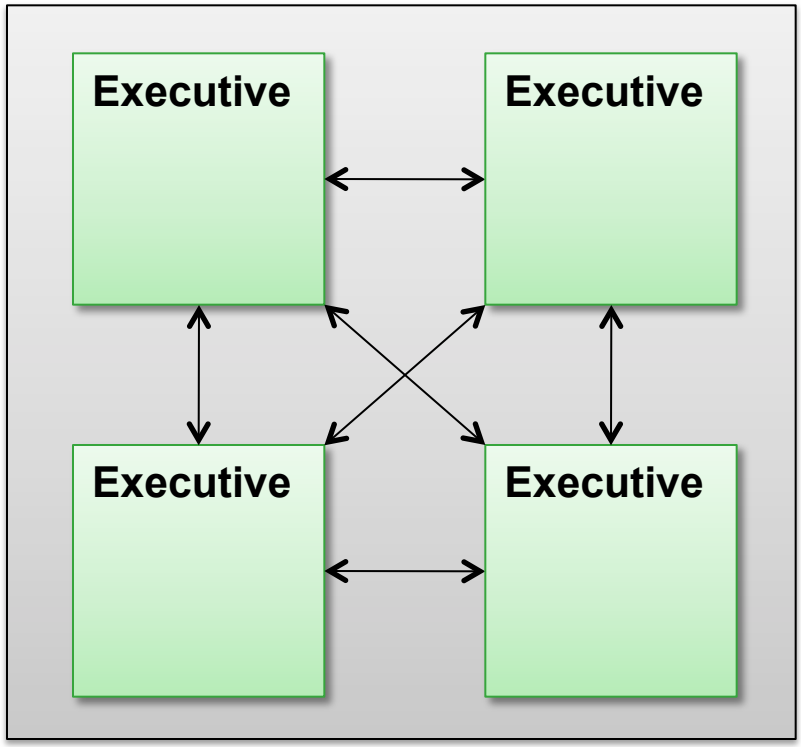
One or more executives per computer contain application and service components



Architecture Foundation (II)

Replicated building blocks

Scalable cluster system architecture



Software Distribution

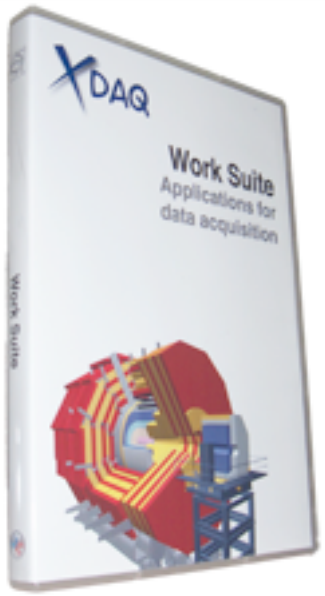
coretools

Core framework

powerpack

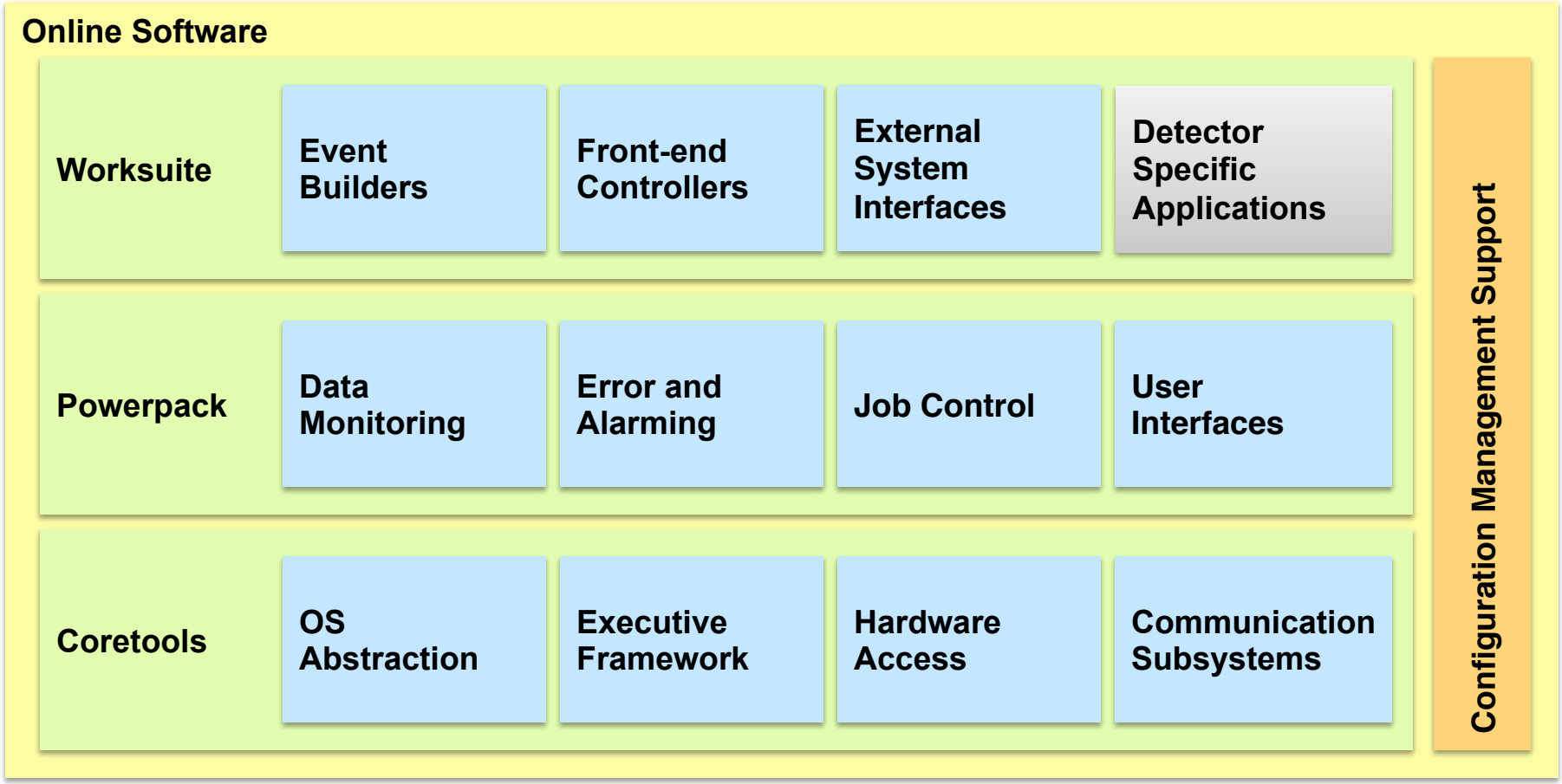
Reusable applications

worksuite

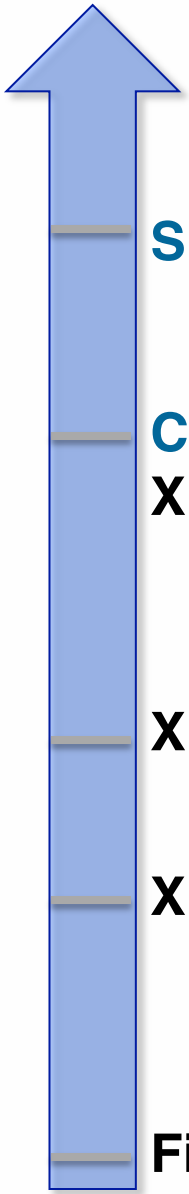


CMS specific applications

Layered View



Timeline



Well consolidated after eleven years
of development and use

2011 — **Successefully used** in the first tree years of LHC

2008 — **Commissioning and first beam event** successfully achieved
XMAS – monitoring, orthogonal to applications

2006

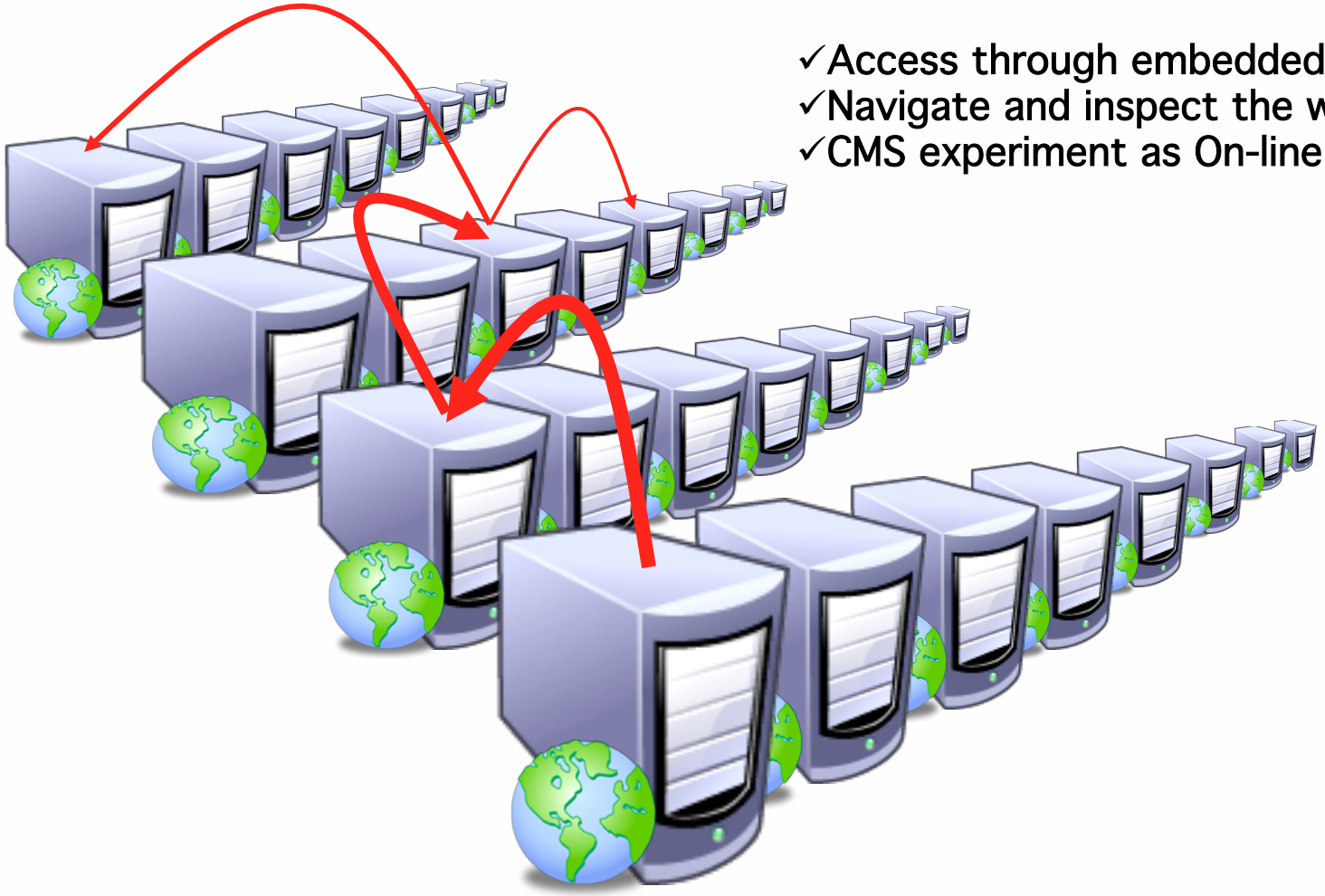
2004 — XDAQ 3 – **experiment wide adoption**

2002

2000 — XDAQ 2 – Web enters DAQ (SOAP)

First version of XDAQ - I_2O communication kernel

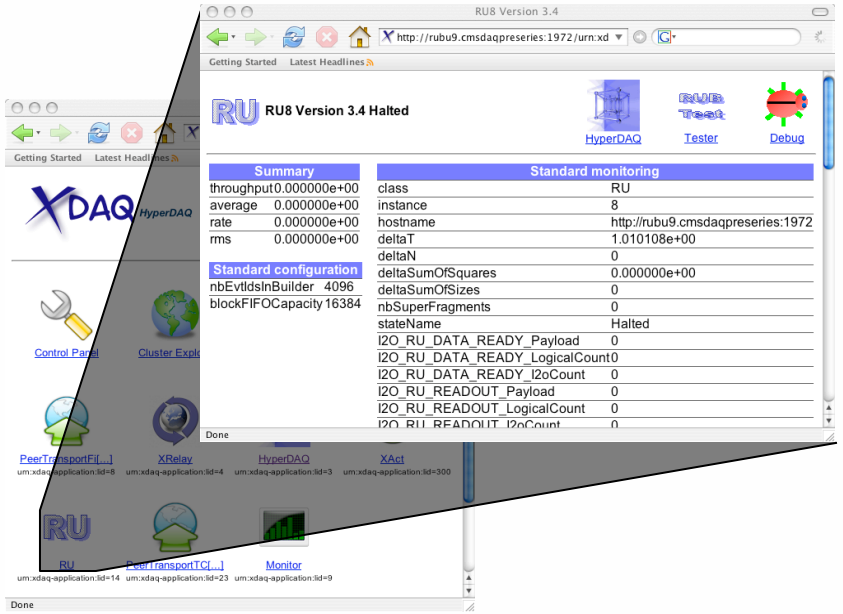
HyperDAQ

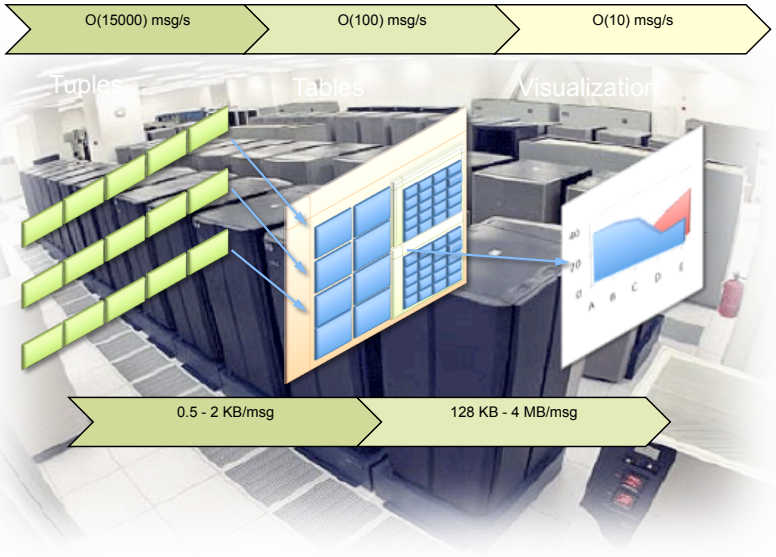


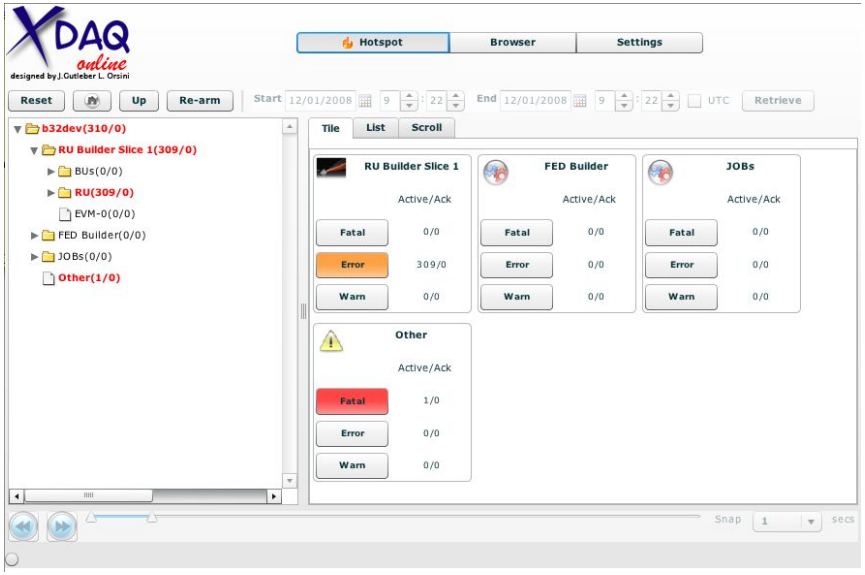
- ✓Access through embedded HTTP
- ✓Navigate and inspect the whole cluster
- ✓CMS experiment as On-line Wide Web

Outlook

- Integrated web technologies
- Monitoring
- Errors and Alarms
- Reusable Event Builder

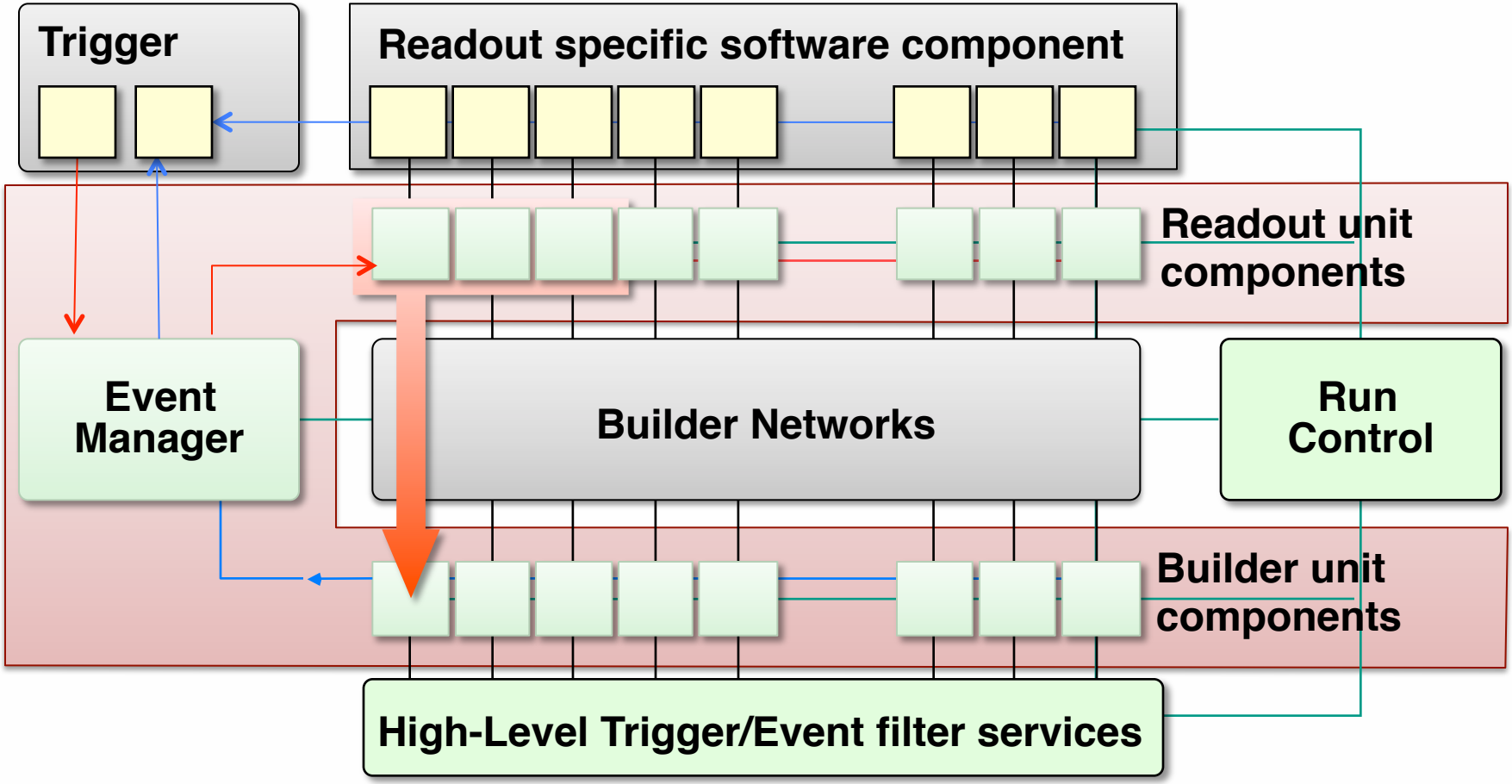






Generic Event Builder

- Configurable in size and network technology
- Customization at boundaries through pluggable components



Configuration Management

- Identification
  - definition of packages
  - versioning
- Traceability
  - Status accounting
  - Documented change history
  - Tickets
- Planning
  - Milestones
  - Priorities, People
- Release
  - Source and binary releases
  - Parallel releases
  - Upgrades

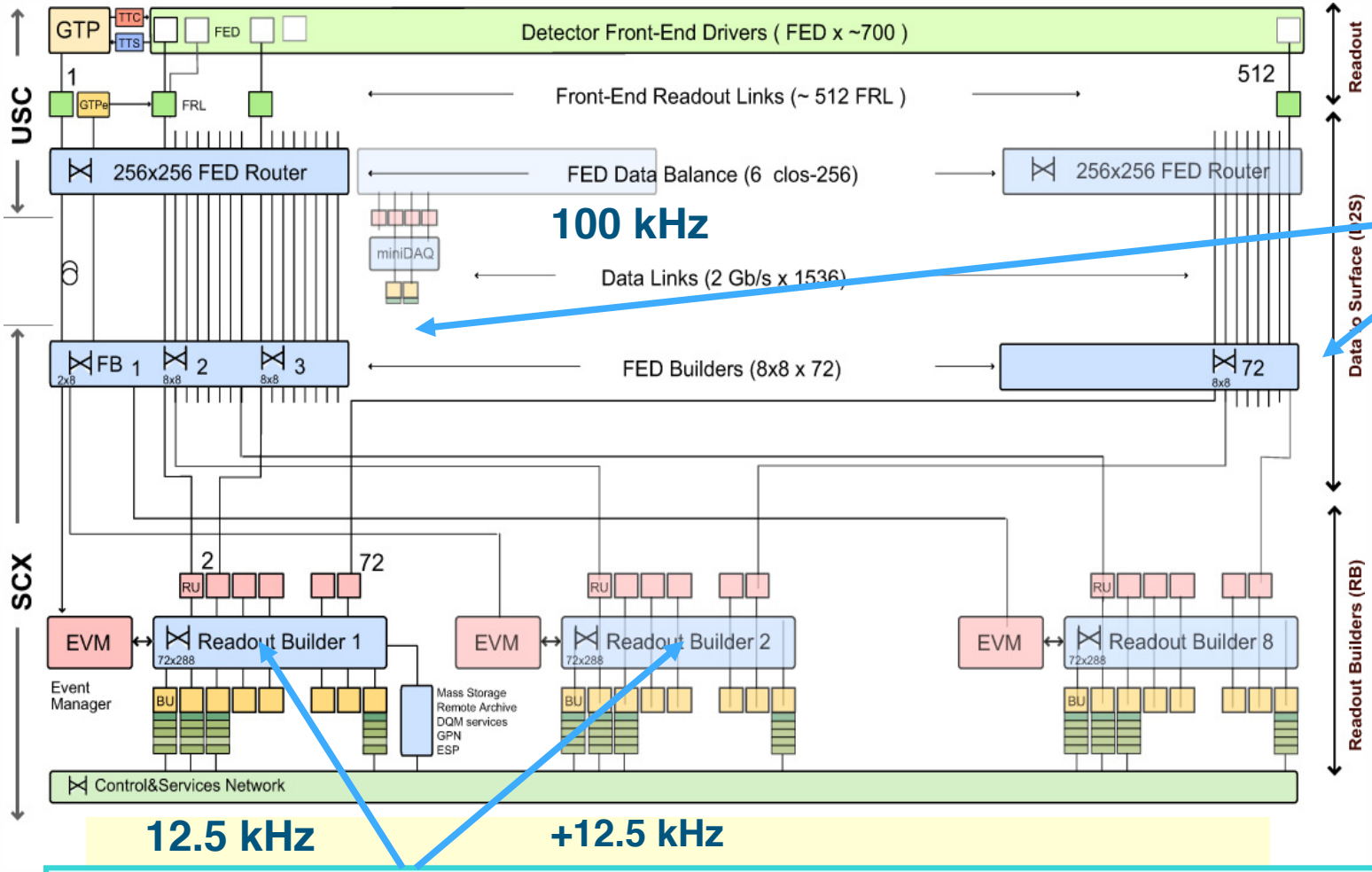
Configuration Management Tools

IT Department

WEB Services

E-Groups

CMS DAQ System



80 FED builders: assemble FED (~2 kB) into super-fragment (~16 kB)

Eight RU builders (EVB DAQ slices):
Assemble super-fragments (~16 kB) into full events (~1 MB) at 12.5 kHz each

CMS DAQ 2011

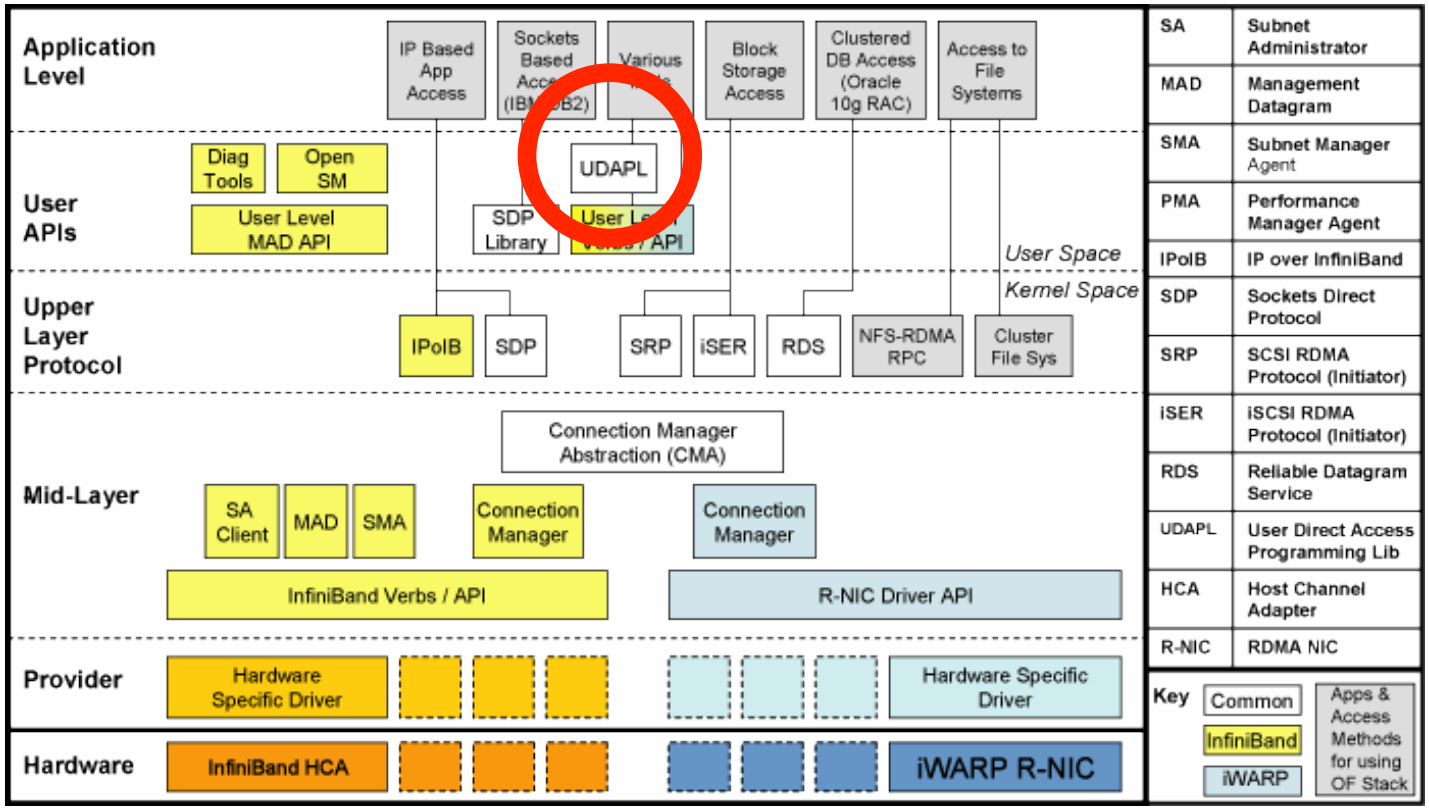
- Full readout of all sub-detectors
- 64 FedBuilders, Myrinet (~1000 applications)
- 8 DAQ Slices, 100 GB/s event builder, Force10 (~1600 applications)
  - Slice with 80 RUs x 125 BUFUs – TCP/IP over GE
- Event Filter (~8200 instances):
  - 720 8-core PCs (Intel Xeon E5430- 2.66 GHz, 16 GB):
    - ~5000 instances of CMSSW-HLT
    - ~100ms/evt
  - 288 8-core PCs (Intel Xeon X5650 2.67 GHz, 24 GB)
    - ~3200 instances of CMSSW-HLT
    - ~81ms/evt
- Storage Manager (16 instances): 2 GB/s, 250 TB buffer.
- CMS on-line farm ~2500 PCs

InfiniBand enabled XDAQ clusters

The OFED Stack (source: OpenFabrics Alliance)

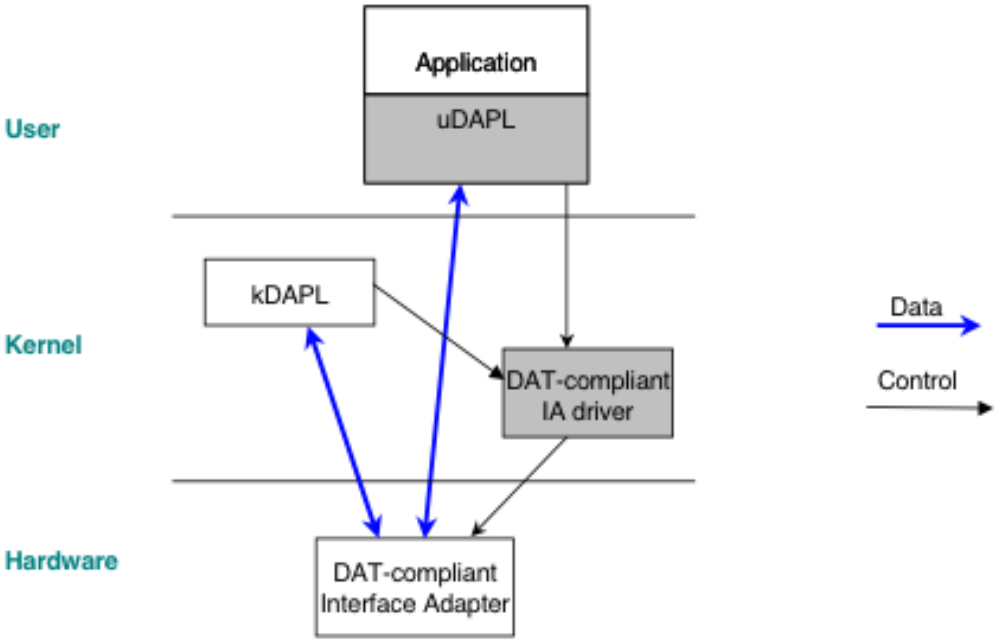
A unified, cross-platform, transport-independent software stack for RDMA and kernel bypass

- http://www.openfabrics.org/



DAT Model (source DAT Collaborative)

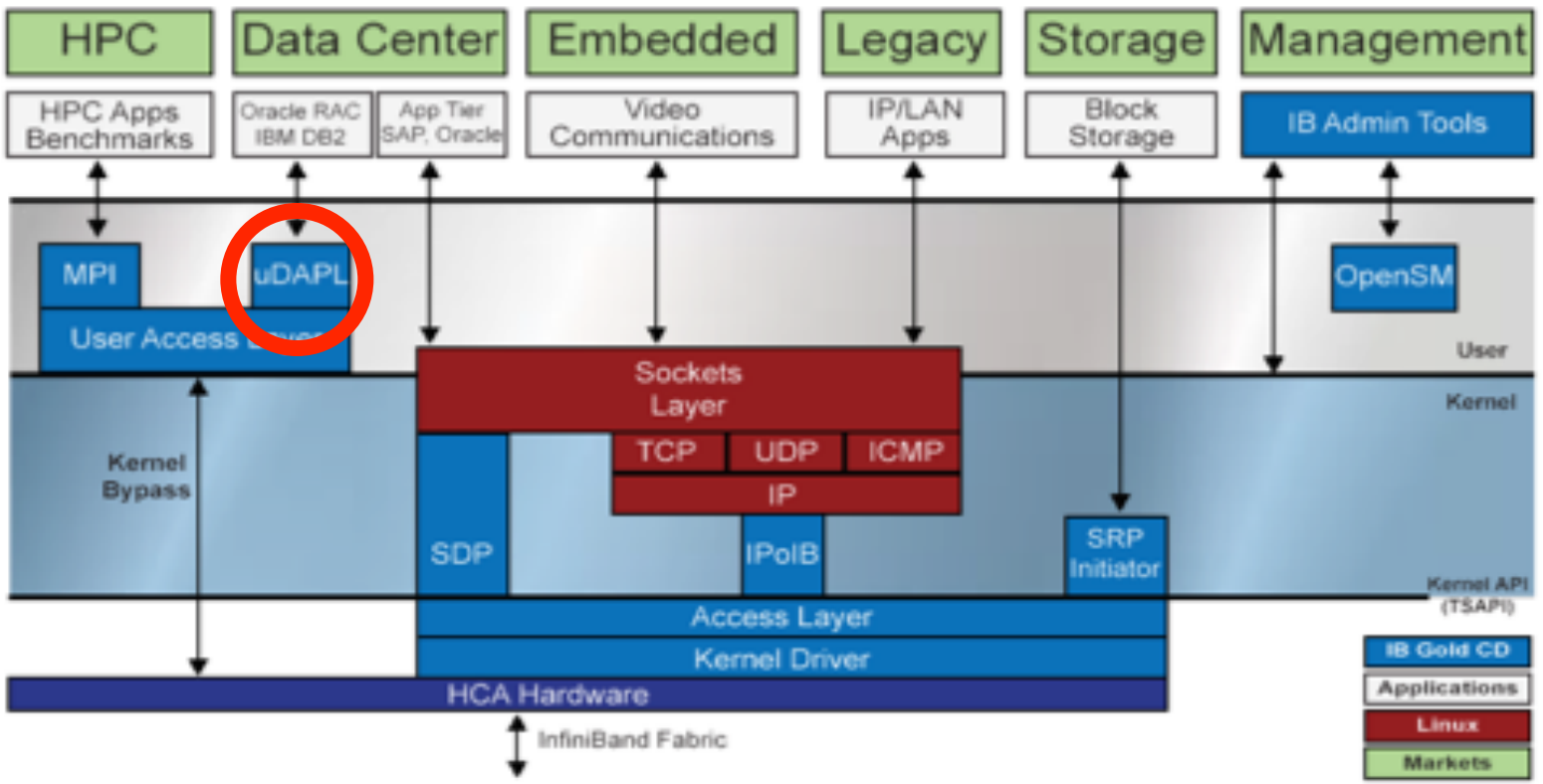
- Developed by DAT collaborative
  - ❑ http://www.datcollaborative.org/
- Transport and platform (OS) independent
- Define user (uDAPL) and kernel (kDAPL) APIs
- DAT supports reliable connection
- Data Transfer Operations send, receive, rdma_read, rdma_write
- uDAPL Version 2.x, January, 2007



Other APIs

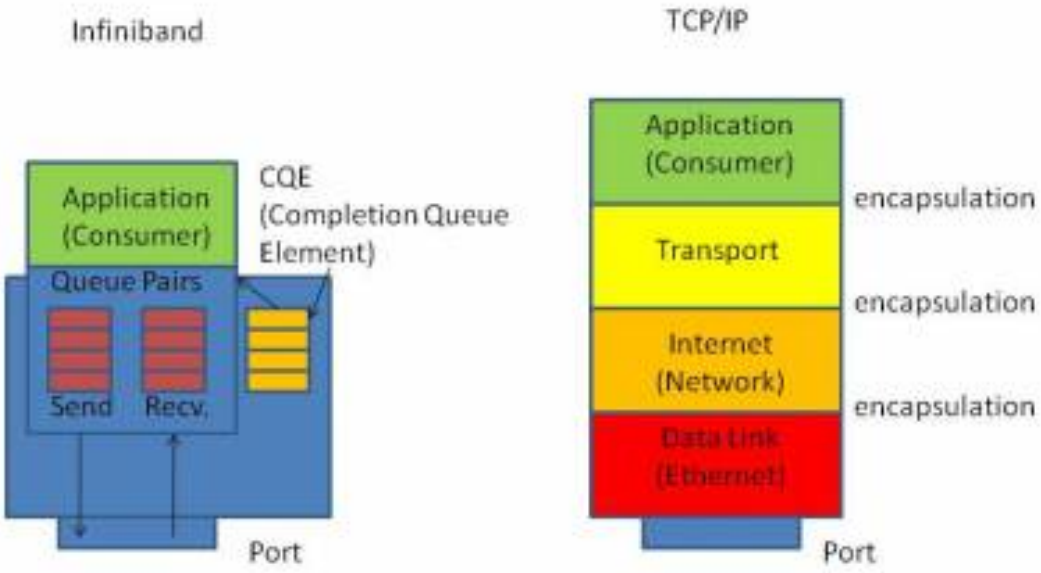
- IPoIB (IP over Infiniband)
- MPI (Message Passing Inferface)
- SDP (Socket Direct Protocol)
- RDS (Reliable Datagram Sockets)

Protocol Stack View



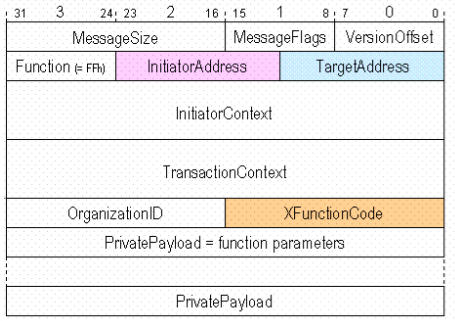
Comparison of the Stacks

The protocol is defined as a very thin set of zero copy functions when compared to thicker protocol implementations such as TCP/IP



Pluggable pt*u*DAPL

- I_2O messaging on InfiniBand transparent to XDAQ applications



XDAQ IB Early Adopters

- XDAQ evaluation at GSI for FAIR (Facility for Antiproton and Ion Research) by J. Adamczewski-Musch, H.G. Essel, S. Linev at GSI Helmholtzzentrum für Schwerionenforschung GmbH, Darmstadt
  - https://www.gsi.de/documents/DOC-2006-Oct-34-1.pdf
- Implemented very first prototype for XDAQ peer transport ptDAPL for I_2O
- Many Thanks to GSI people
  - Their interest in XDAQ technology
  - Extremely useful contribution for enabling XDAQ with Infiniband
  - Shorten learning curve for new technology
  - Reference measurements for comparison

LHCb IB Collaboration

- Infiniband Event-Builder Architecture Test-beds for Full Rate Data Acquisition in LHCb – CHEP 2010
  - ❑ http://cdsweb.cern.ch/record/1302037/files/LHCb-TALK-2010-151.pdf

- Many thanks to LHCb people (Niko Neufeld, Jean Christophe Garnier)
  - Provide access to their network cluster with IB installation ready to use
  - Reference measurements for comparison

LHCb IB Environment (2010)

- 8 nodes
- Qlogic 12300
  - QDR 4X
  - 32 Gb/s
- Processor type Intel Xeon E5520
- Processors x cores x clock (GHz) 2 x 4 x 2.27
- RAM (GiB) 3
- HCA qle7340 4x QDR
- SLC5/64 bit - Kernel 2.6.18
- CMSOS release 11
  - GEVB (Generic Event Builder)
  - PTuDAPL (XDAQ Peer Transport uDAPL)

XDAQ IB pt*u*DAPL

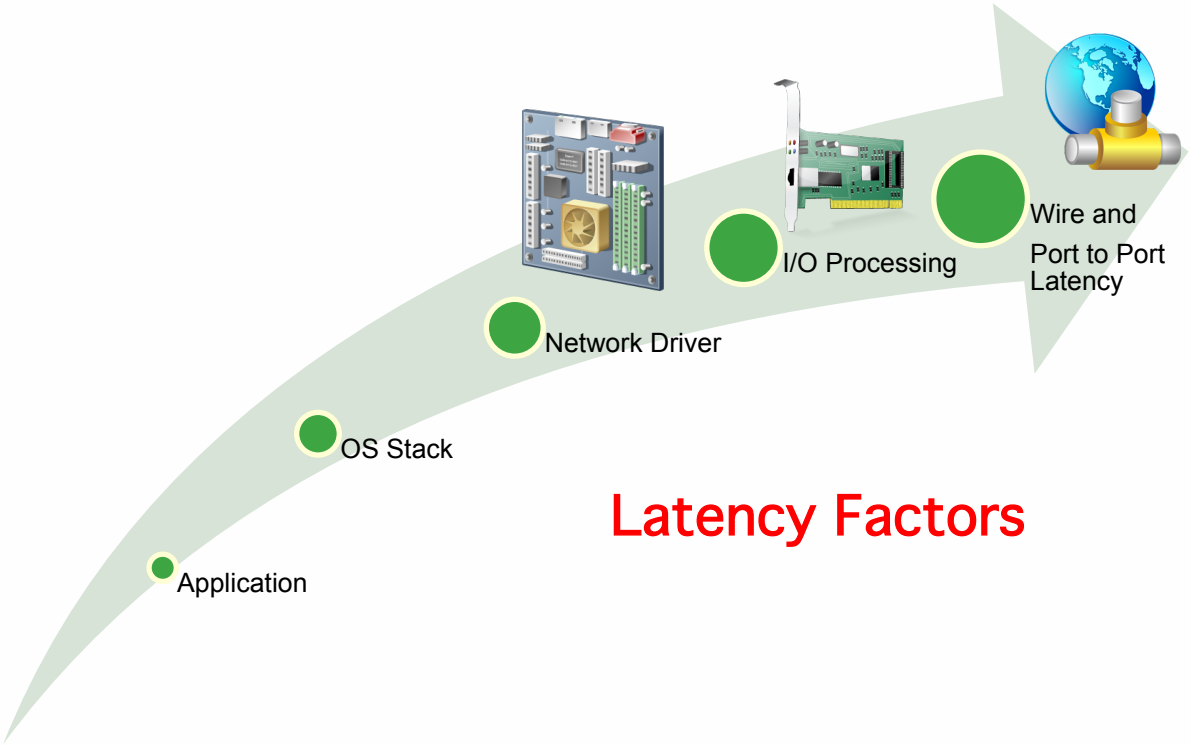
- Reverse engineering of GSI ptDAPL
  - Identified overhead in buffer management (virtual mapping for all packets in use for all send/recv operations)
  - Implementation based on non blocking TCP peer transport not effective for use of uDAPL API
  - Not optimized workloop usage
  - Based on DAT Spec 1.x
- Full re-factoring into new pt*u*DAPL
  - Use of smart memory pool based on uDAPL memory region allocator (random access to memory with no intermediate management by using cookies)
  - Profiting for inherent non blocking and queuing of uDAPL API for minimizing latency (removed intermediate output fifo)
  - Several optimizations in workloops, wrapper classes etc.
  - All I/O operations centered on dedicated uDAPL memory pool
  - Based on DAT Spec 2.x

Reference XDAQ Applications

- Roundtrip
  - Used to measure latency
- MStreamIO
  - Used to measure throughput
- Event Builders
  - GEVB Generic Event Builder
  - RUBuilder Official CMS event builder

Roundtrip

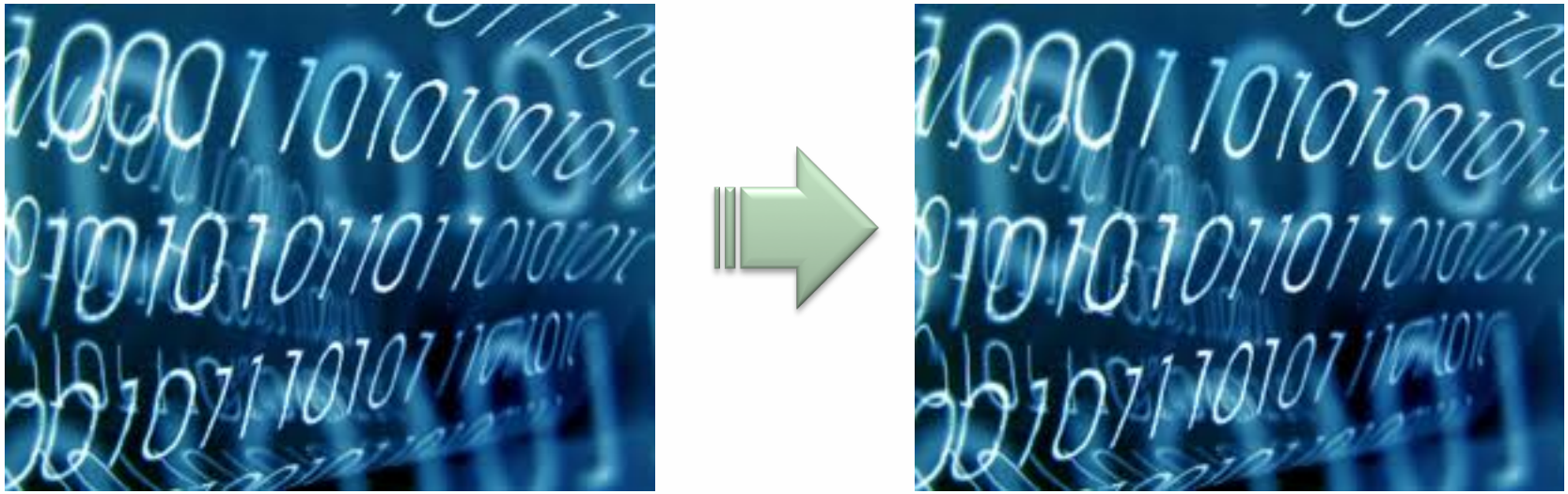
- Simple XDAQ application to compute the One-way delay
- Time packet to travel from a specific source to a specific destination and back again
- One-way latency is measured by timing a round-trip message and dividing the obtained result by two



Stream I/O

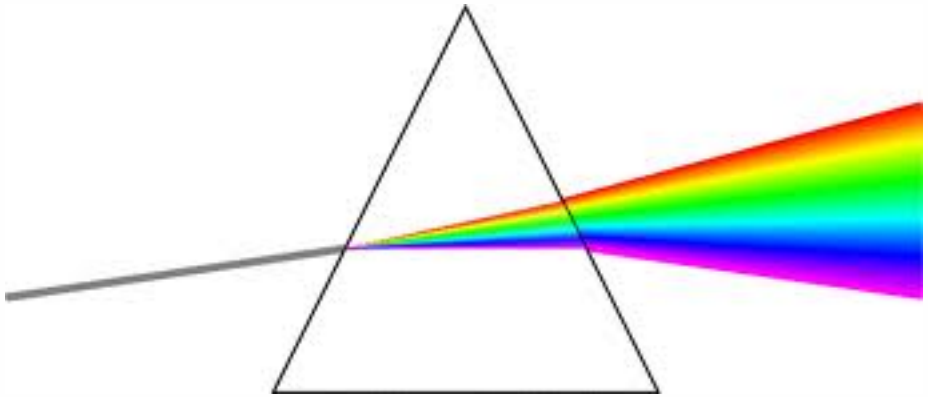
- Unidirectional throughput (bandwidth) is measured using a unidirectional send of N messages. Time sampling is done at the receiver side and starts with the first incoming message

- Bidirectional throughput measures the case where both of the two hosts are busy sending data to each other. Note that the theoretical maximum here is twice the link speed for full duplex media



Event Builders

- RUBuilder
  - Currently used in central DAQ
  - Dataflow protocol 5 hops (allocate, confirm, send, cache and release)
- GEVB
  - Same functionalities and interfaces as RUBuilder
  - Dataflow control 3 hops (allocate, ship and cache)
  - It was used for sub-detector data acquisition testbeams before TDR and adoption of RUBuilder



Preliminary Measurements

Test Planning

UDAPL

- RTT, I/O
- Throughput 1x1, 1x2, 1x4
- GEVB 1x1, 1x2, 1x4, 2x1, 4x1, 2x2, 3x3
- RUBuilder 1x1, 1x2, 1x4, 2x1, 4x1, 2x2, 3x3

ATCP(10GE)

- **RTT**
- **Throughput 1x1, 1x2, 1x4**
- **GEVB 1x1, 1x2, 1x4, 2x1, 4x1, 2x2, 3x3**
- **RUBuilder 1x1, 1x2, 1x4, 2x1, 4x1, 2x2, 3x3**

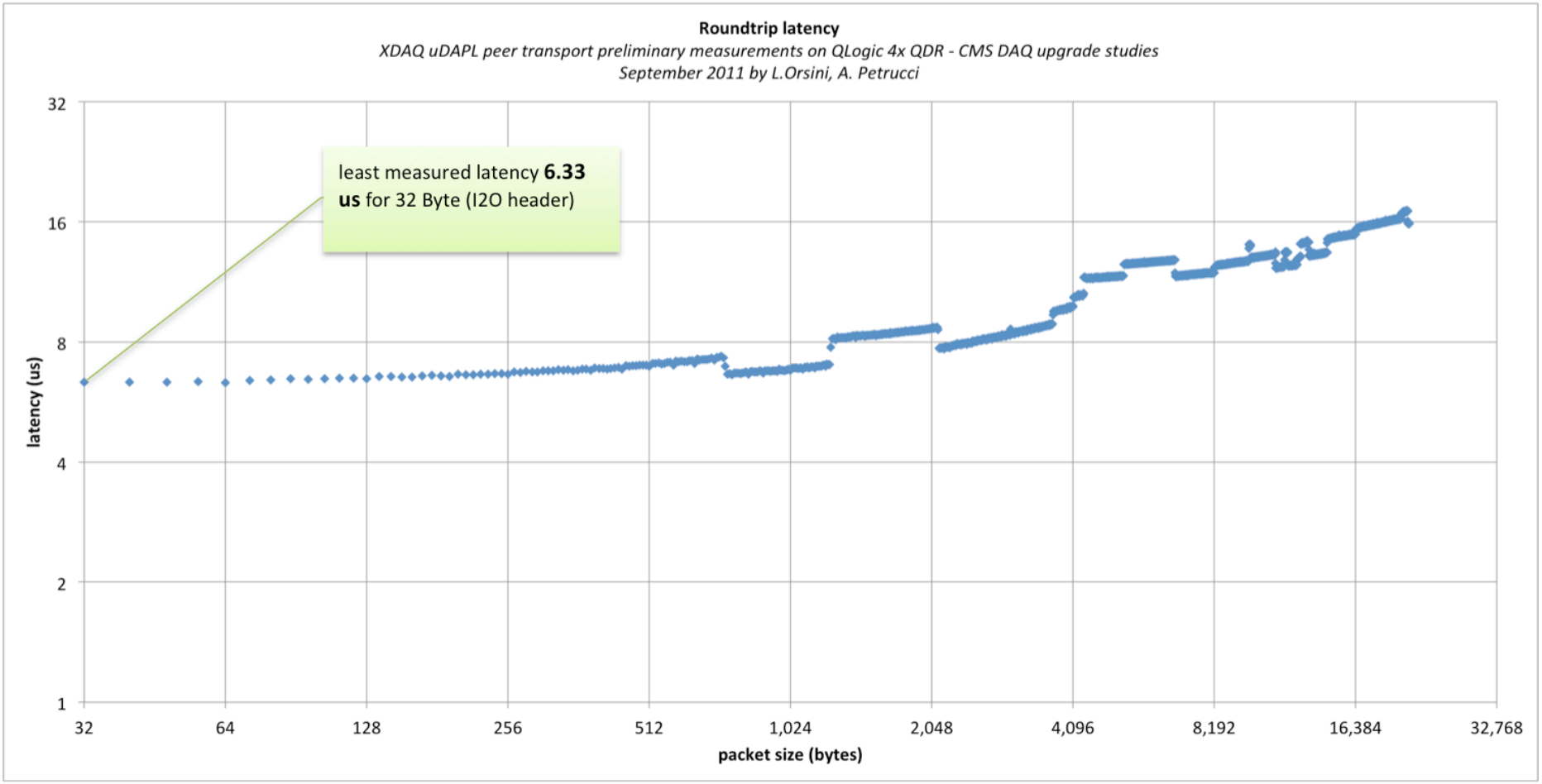
IPoIB

- **RTT,I/O**
- **Throughput 1x1, 1x2, 1x4**
- **GEVB 1x1, 1x2, 1x4, 2x1, 4x1, 2x2, 3x3**
- **RUBuilder 1x1, 1x2, 1x4, 2x1, 4x1, 2x2, 3x3**

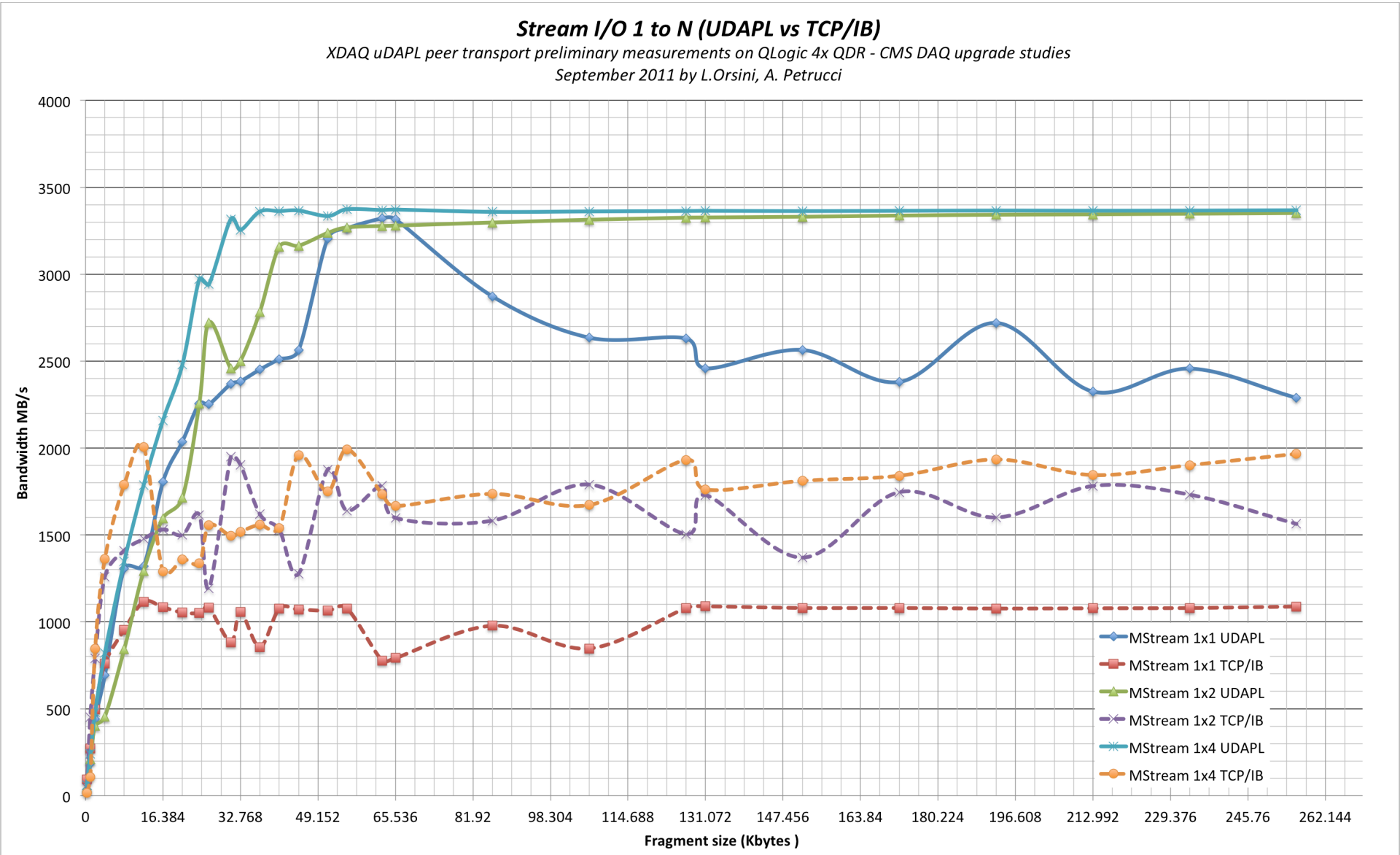
Variable sizes

- **GEVB both uDAPL/IPoIB**
- **RUBuilder uDAPL/IPoIB**

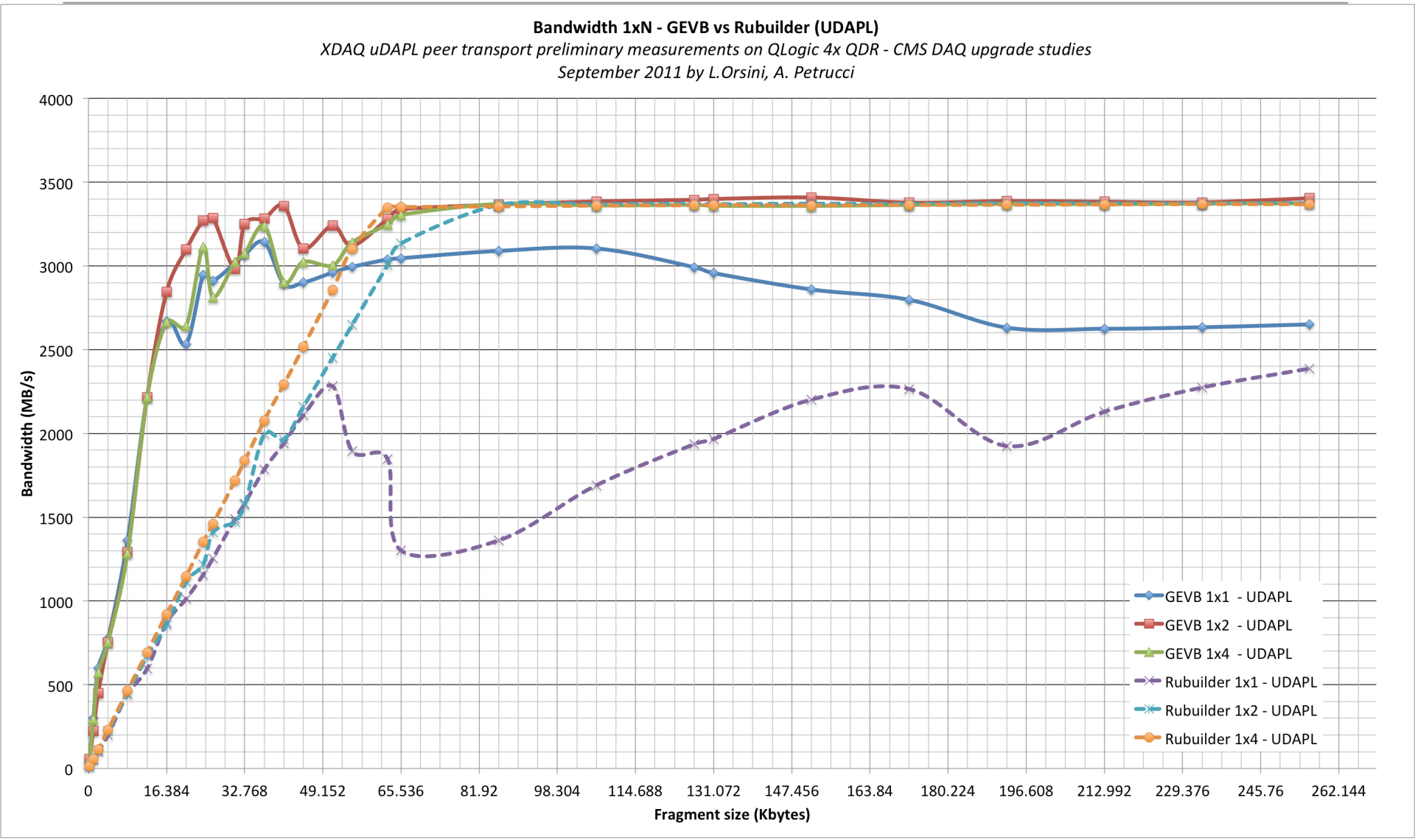
RTT


Roundtrip latency
XDAQ uDAPL peer transport preliminary measurements on QLogic 4x QDR - CMS DAQ upgrade studies
September 2011 by L.Orsini, A. Petrucci
least measured latency 6.33 us for 32 Byte (I2O header)
latency (us)
32
16
8
4
2
1
32
64
128
256
512
1,024
2,048
4,096
8,192
16,384
32,768
packet size (bytes)

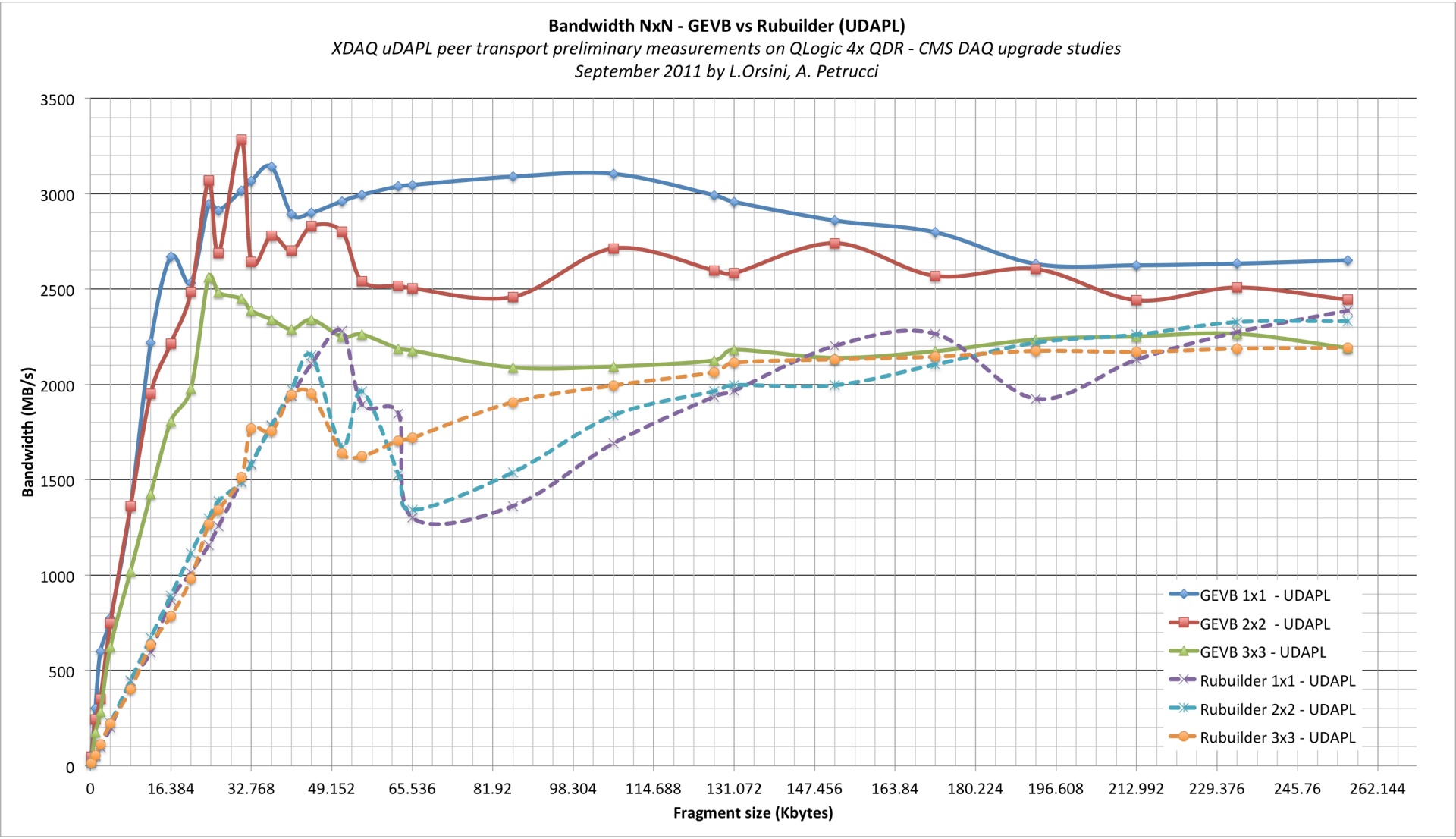
Throughput


Stream I/O 1 to N (UDAPL vs TCP/IB)
XDAQ uDAPL peer transport preliminary measurements on QLogic 4x QDR - CMS DAQ upgrade studies
September 2011 by L.Orsini, A. Petrucci
Bandwidth MB/s
4000
3500
3000
2500
2000
1500
1000
500
0
0
16.384
32.768
49.152
65.536
81.92
98.304
114.688
131.072
147.456
163.84
180.224
196.608
212.992
229.376
245.76
262.144
Fragment size (Kbytes)
MStream 1x1 UDAPL
MStream 1x1 TCP/IB
MStream 1x2 UDAPL
MStream 1x2 TCP/IB
MStream 1x4 UDAPL
MStream 1x4 TCP/IB

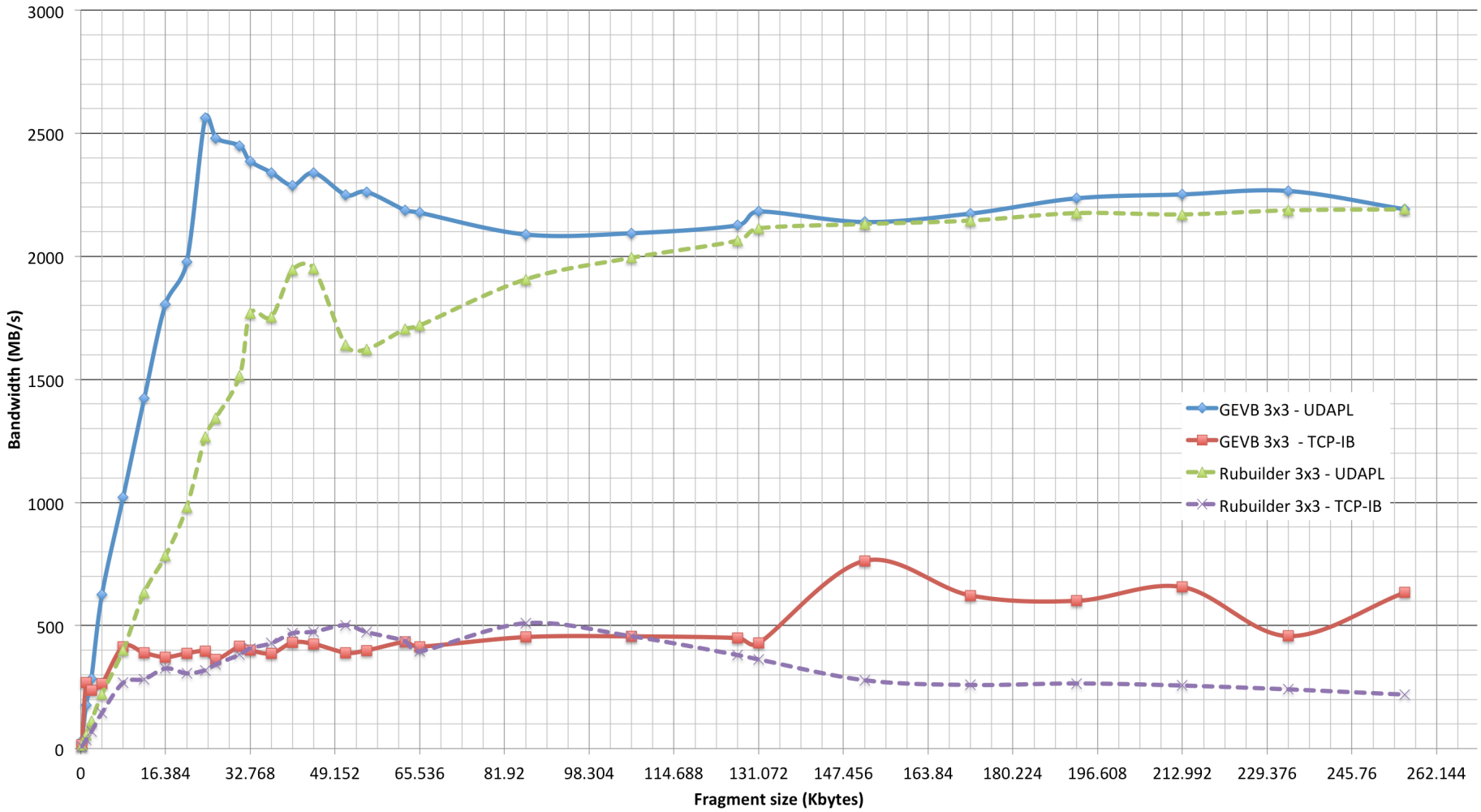
EVB-1xN pt*u*DAPL



EVB-NxN pt*u*DAPL



EVB-3x3 pt*u*DAPL vs pt*a*TCP



Preliminary results

- Use rubuilder off-the-shelf with both peer transport
- Use default configuration for both DAPL and TCP
- Latency pt/*u*DAPL (RTT) 6.33 us
- pt/*u*DAPL over 3 GB/S with packet size greater than 26 Kbytes
- pt/atcp reached 2.4 GB/s with packet size greater than 172 Kbytes
- Largest size of event builder 3x3 (6 nodes + 1 node for event manager)
  - maximum bandwidth 2.5 GB/s at 24 Kbytes with pt/*u*DAPL
  - maximum bandwidth 762 MB/s at 150 Kbytes with pt/atcp

Conclusion

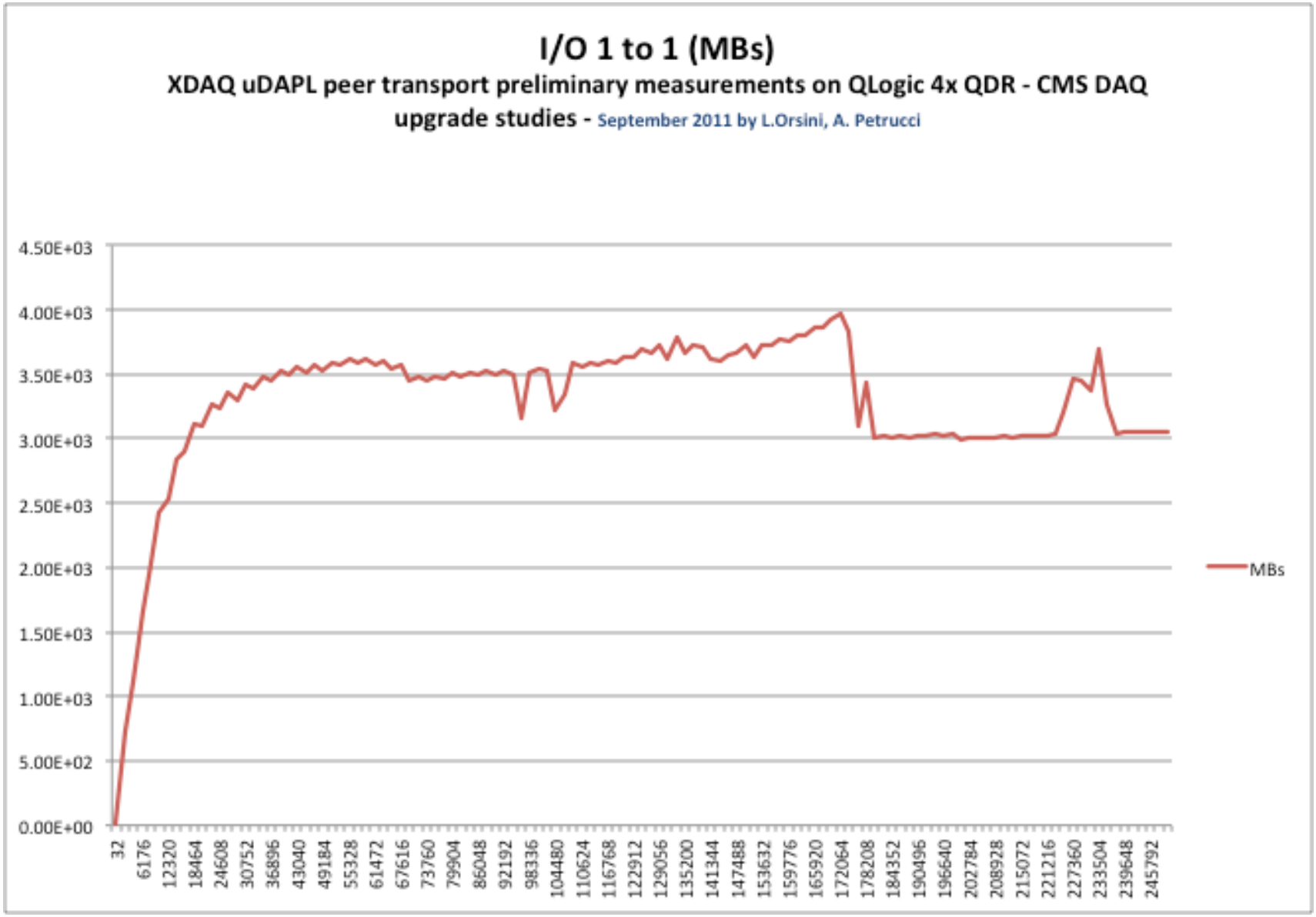
- Building a DAQ system as a process of **assembly re-usable components** in a predetermined way **rather than** a **programming** task.

- Achieved **a uniform DAQ product-line** for all CMS data acquisition application scenarios ranging from single CPU setups to the final systems comprising thousands of nodes.

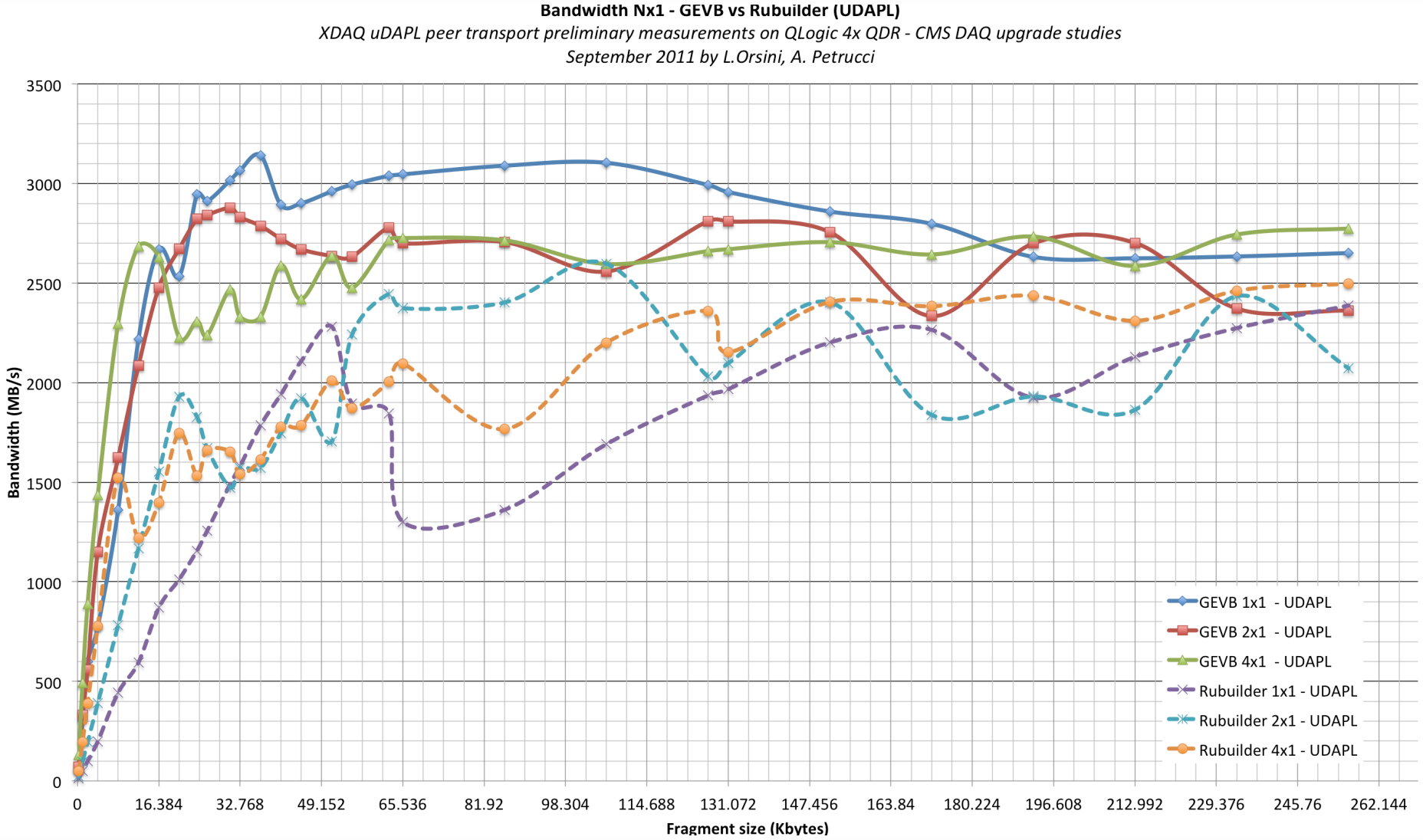
http://svnweb.cern.ch/trac/cmsos

Backup

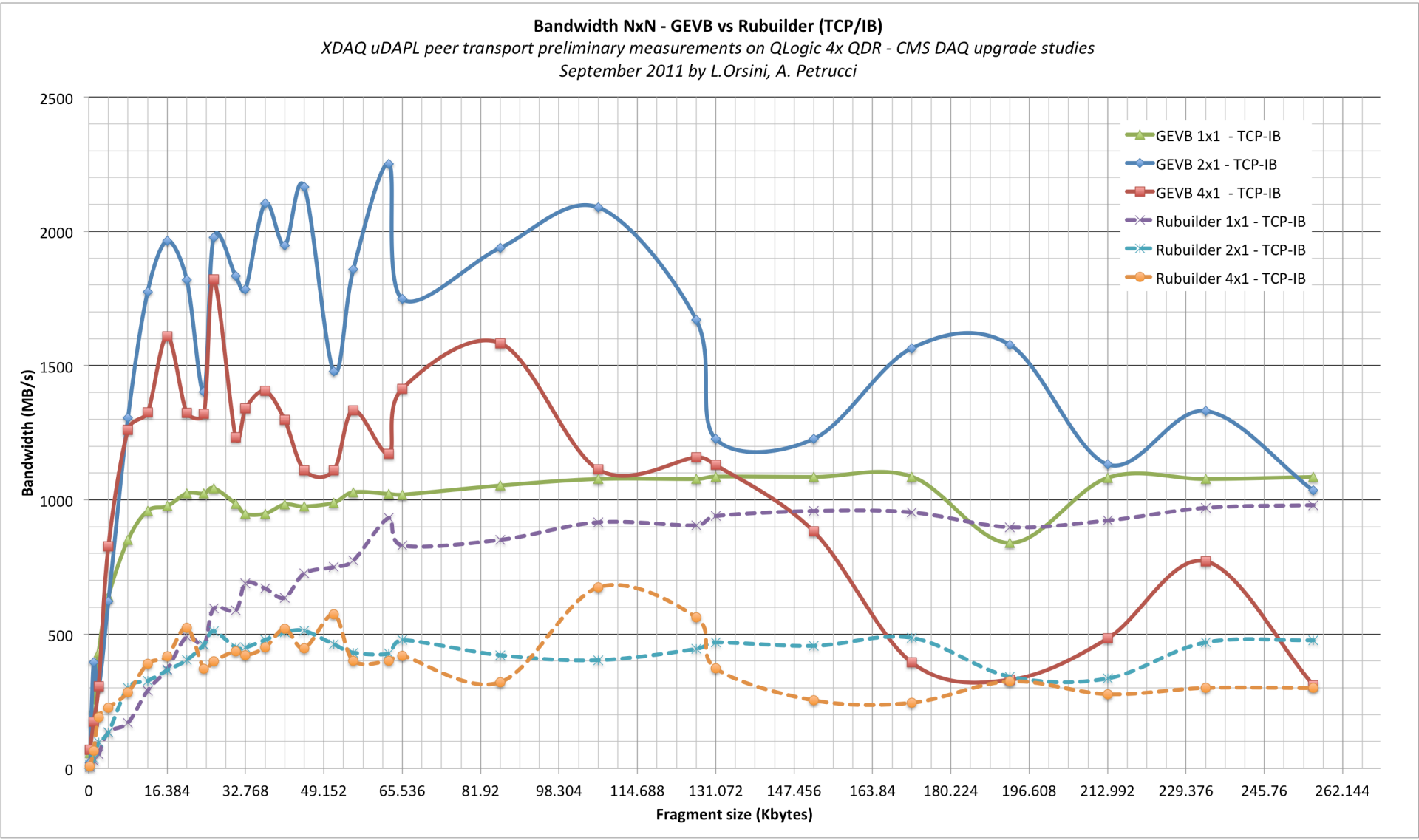
I/O


I/O 1 to 1 (MBs)
XDAQ uDAPL peer transport preliminary measurements on QLogic 4x QDR - CMS DAQ upgrade studies - September 2011 by L.Orsini, A. Petrucci
4.50E+03
4.00E+03
3.50E+03
3.00E+03
2.50E+03
2.00E+03
1.50E+03
1.00E+03
5.00E+02
0.00E+00
32
6176
12320
18464
24608
30752
36896
43040
49184
55328
61472
67616
73760
79904
86048
92192
98336
104480
110624
116768
122912
129056
135200
141344
147488
153632
159776
165920
172064
178208
184352
190496
196640
202784
208928
215072
221216
227360
233504
239648
245792
MBs

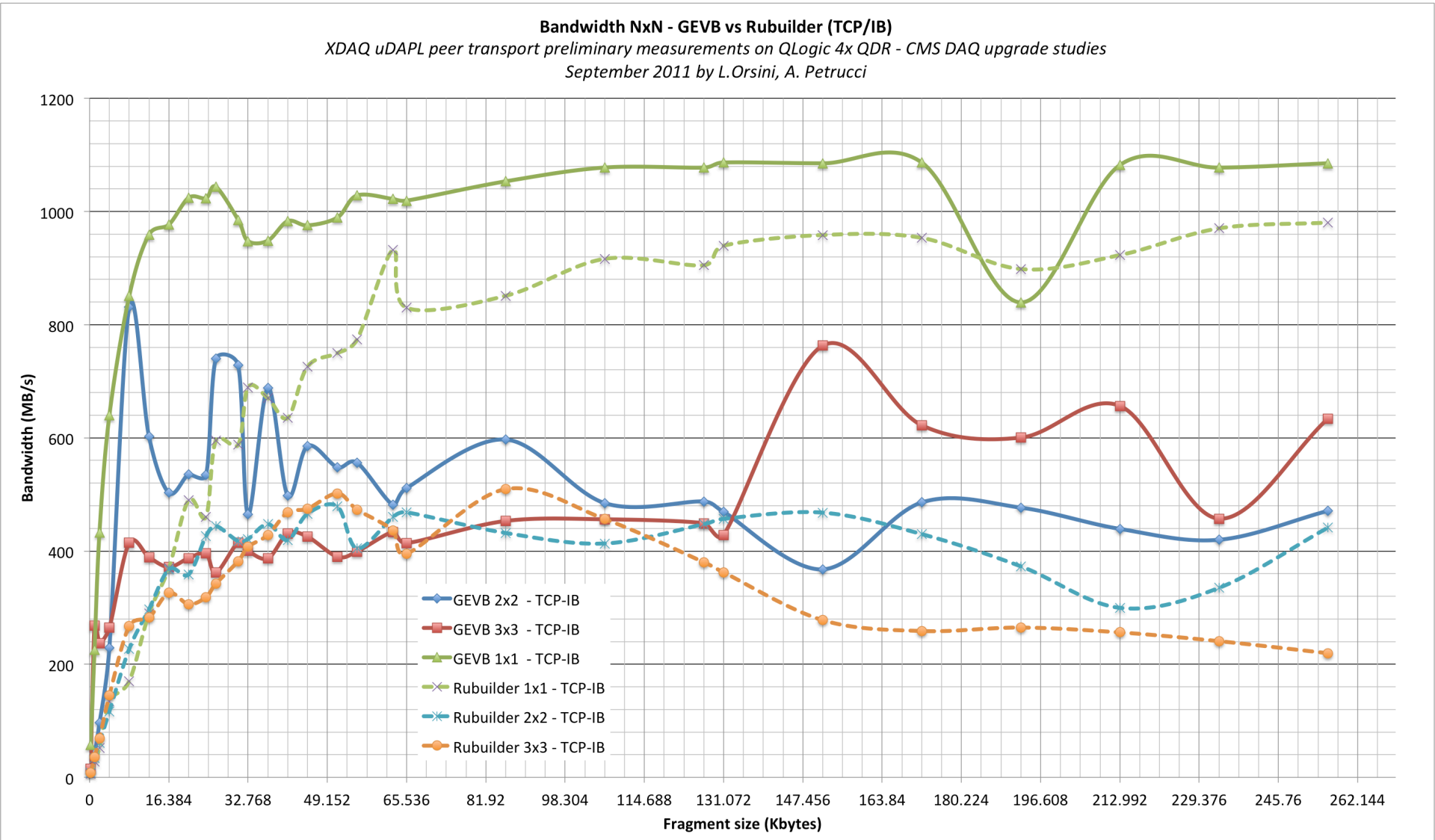
EVB-Nx1 pt*u*DAPL



EVB-Nx1 ptaTCP



EVB-NxN ptaTCP



EVB-1xN ptaTCP