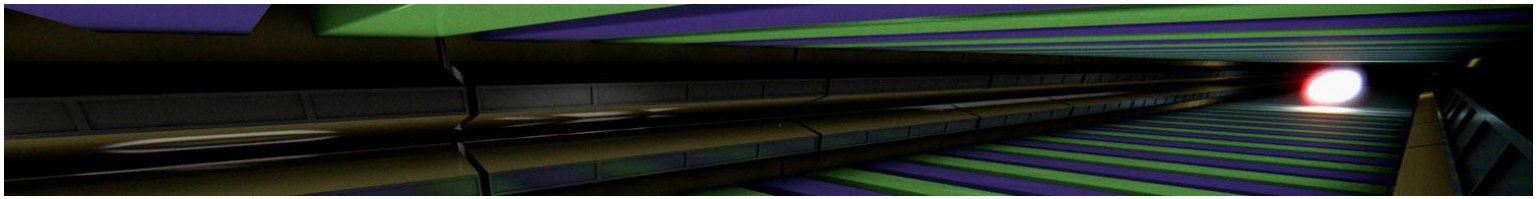


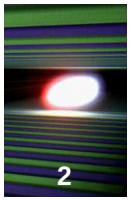
10GE network tests with UDP

Janusz Szuba

European XFEL

European XFEL

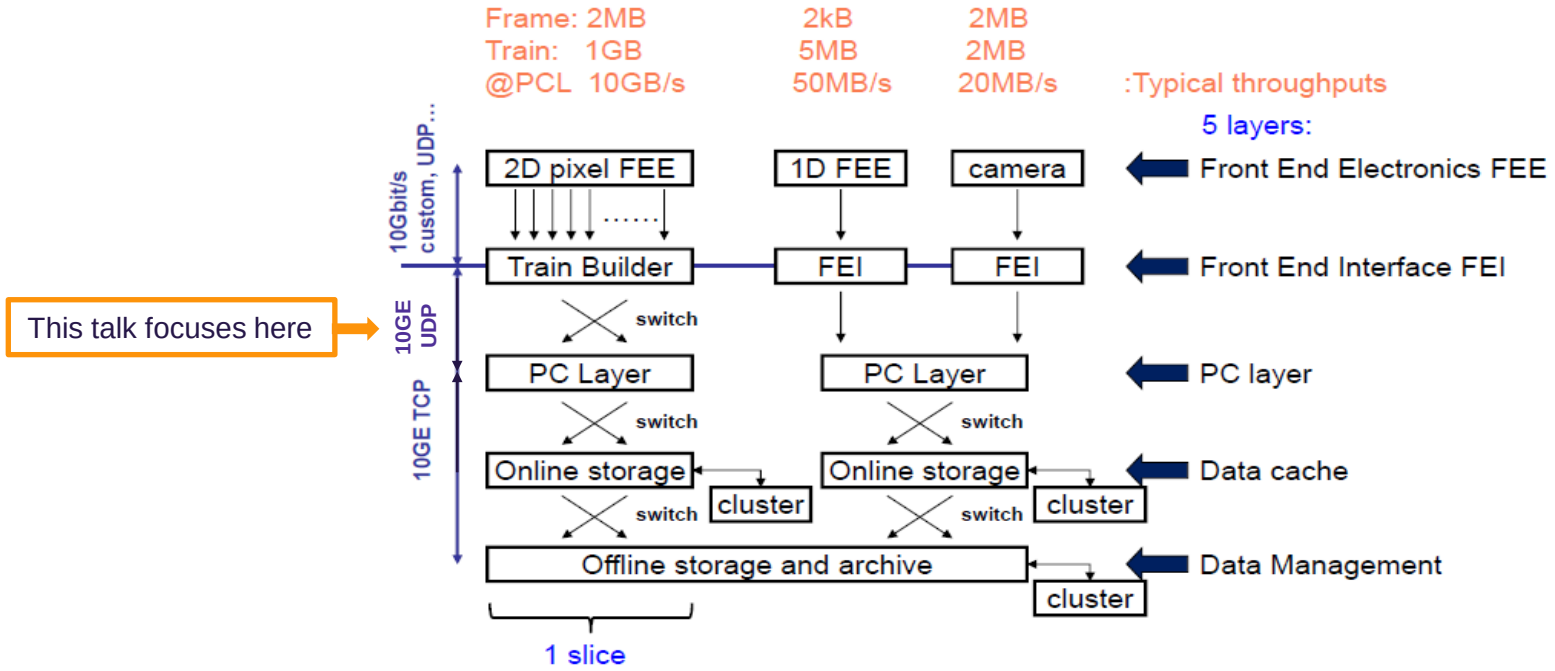
Outline



- Overview of initial DAQ architecture
- Slice test hardware specification
- Initial networking test results
- DAQ software UDP tests
- Summary

European XFEL

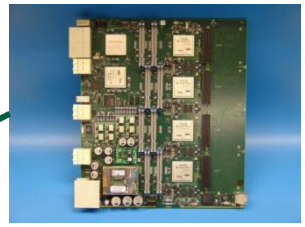
Initial DAQ architecture



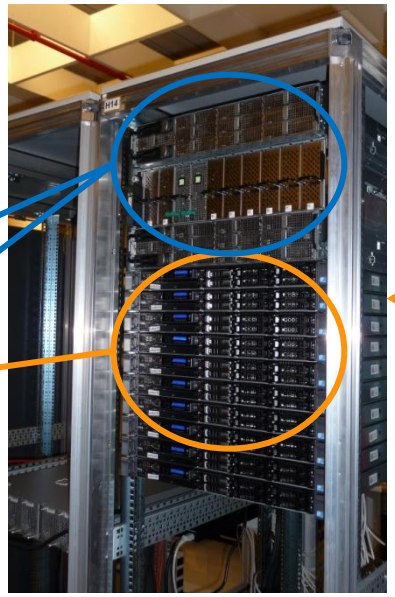
- Multiple layers with well defined APIs
  - to allow insertion of new technology layers
- Multiple slices for partitioning and scaling
  - camera sizes will increase and slice replication will be one solution
- Enforce data reduction and rejection in all layers
  - early data size reductions and data rejection are needed to reduce storage resources

European XFEL

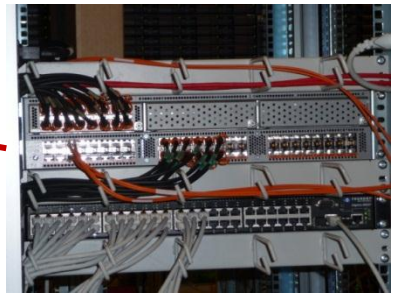
Slice test hardware setup



TB prototype

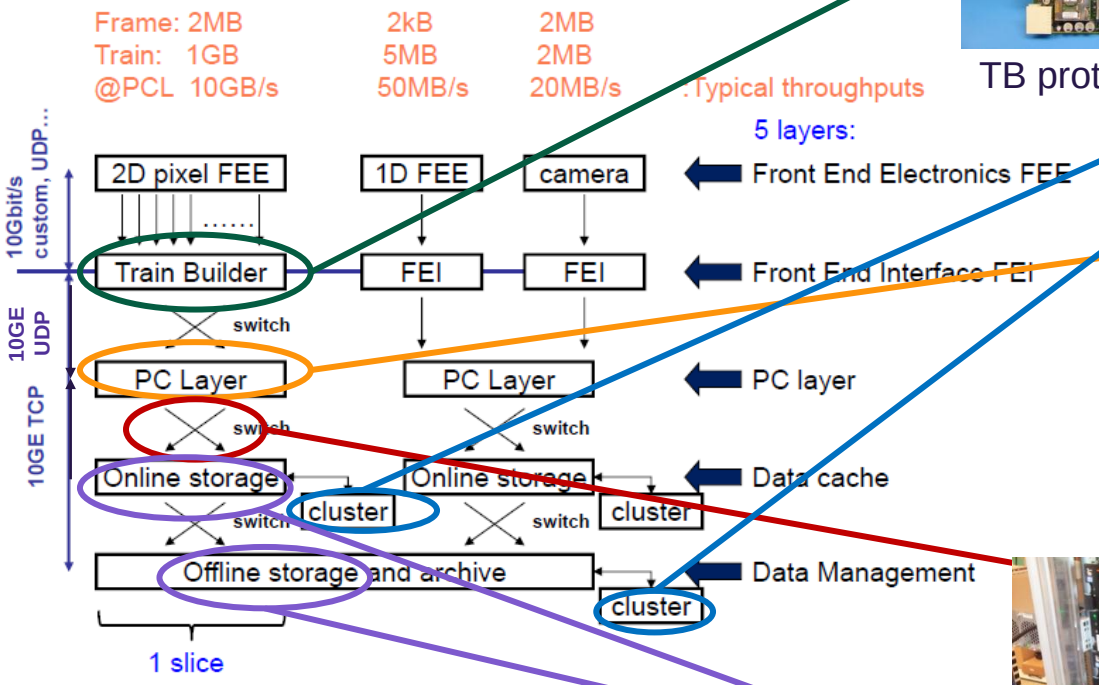


PC and compute nodes



10 Gbps switch

Storage servers



European XFEL

Hardware specification

- Final width of slice is 16 inputs from TB to 16 PCL servers – covering 1Mpxl detector
- Specification of the first ½ of slice:

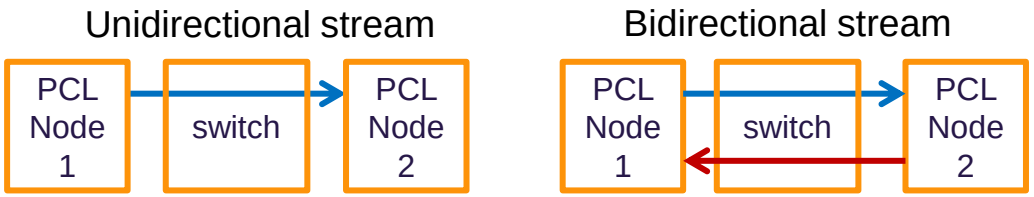
h/w	function	#	Spec
PowerEdge R610	PC layer	8	3GHz, 2Socket, 6Core, Intel X5675, 96GB, Intel X520 DA2 dual 10GbE NIC
PowerEdge R610	Metadata server	2	3GHz, 2Socket, 6Core, Intel X5675, 96GB, Intel X520 DA2 dual 10GbE NIC
PowerEdge C6145	Computing cluster	2	2.6GHz, 4Socket, 16Core, AMD 6282SE, 192GB, Intel X520 DA2 dual 10GbE NIC
Cisco 5596UP	10GbE switch	1	48 ports (extendible to 96), wire speed, SFP+
PowerEdge C410x	PCIe chassis	1	8 HIC Inputs,16 x16 PCIe Sleds
NVIDIA M2075	Computing	4	PCIe x16 GPGPU
IBM x3650M4	9TB servers (fast SAS)	6	2.6GHz, 2Sock, 8Core, Intel E5-2670 SandyBridge, 128GB, Mellanox ConnectX-3 VPI, PCIe Gen3, QSFP (with adapter to SFP+) Dual NIC, 14x 900GB 10Krpm 6Gbps SAS, RAID6
IBM x3650M4	Servers for extension chassis	2	2.6GHz, 2Sock, 8Core, Intel E5-2670 SandyBridge, 128GB, Mellanox ConnectX-3 VPI, PCIe Gen3, QSFP (with adapter to SFP+) Dual NIC
IBM EXP2512	28TB extension chassis	2	3TB 7.2Krpm 6Gbps NL SAS, RAID6, connected through 2 mini SAS connectors
Demonstrator	TB	2	Board and ATCA crate

Initial network tests with system tools

- Two hosts (1+2), each dual 10Gbps NIC, connected via switch
- Tools
  - netperf
    - netperf -H 192.168.142.55 -t tcp_stream -l 60 -T 3,4 -- -s 256K -S 256K
  - mpstat for CPU and interrupts usage

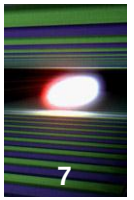
- Settings
  - Socket buffer size (128kB-4MB)
  - Adjusted kernel parameters
  - MTU 1500 and 9000 (Jumbo frames)
  - CPU affinity
  - 2 modes:

```
net.core.rmem_max = 16777216          # 131071
net.core.wmem_max = 16777216          # 131071
net.core.rmem_default = 10000000      # 129024
net.core.wmem_default = 10000000      # 129024
net.core.netdev_max_backlog = 300000  # 1000
Txqueuelen = 10000                    # 1000
```



European XFEL

TCP and UDP throughput tests



- Both TCP and UDP bandwith at wire speed

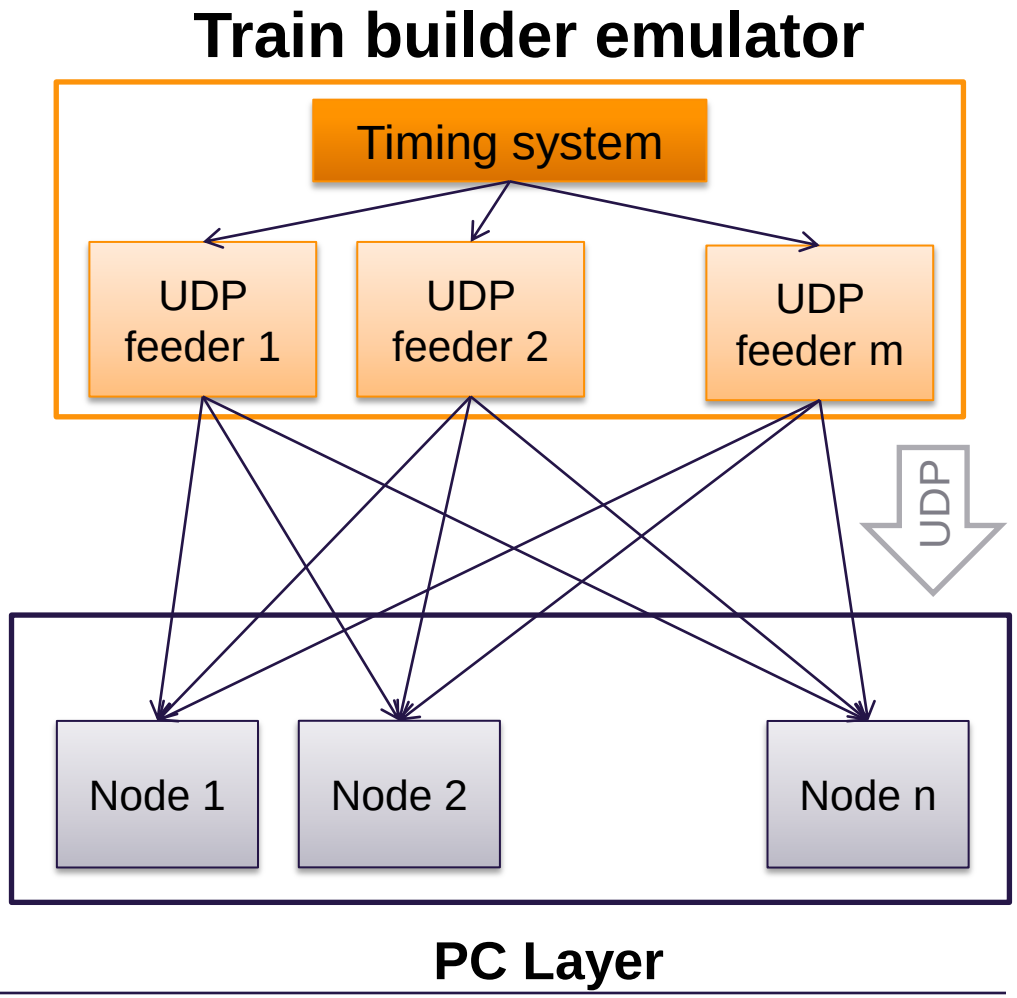
TCP MTU 9000	send1	recv2	send2	recv1	send1	recv2	send2	recv1
buffer size B	512k	512k	512k	512k	1M	1M	1M	1M
bw Mb/s	9796	9796	9777	9777	9899	9899	9899	9899
cpu sys %	27	19	28	22	25	19	26	20
cpu soft %	2	9	2	9	4	8	4	9
intr/s	8k	8k	8k	8k	8k	8k	8k	8k

UDP MTU 9000	send1	recv2	send2	recv1	send1	recv2	send2	recv1
buffer size B	512k	512k	512k	512k	1M	1M	1M	1M
bw Mb/s	9926	9927	9927	9927	9927	9927	9927	9927
packet loss %	$<10^{-6}$		$<10^{-6}$		$<10^{-6}$		$<10^{-6}$	
cpu sys %	27	26	28	25	26	26	28	25
cpu soft %	5	0.1	5	0.1	6	0.1	5	0.1
intr/s	8k	8k	8k	8k	8k	8k	8k	8k

- More tuning required to reduce UDP packet loss
- Further tests performed with developed DAQ software

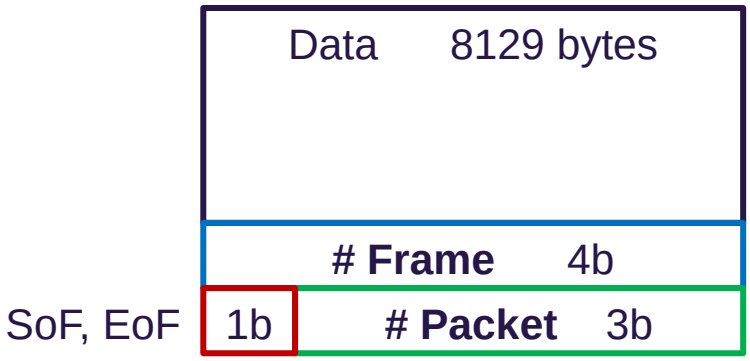
Software architecture - overview

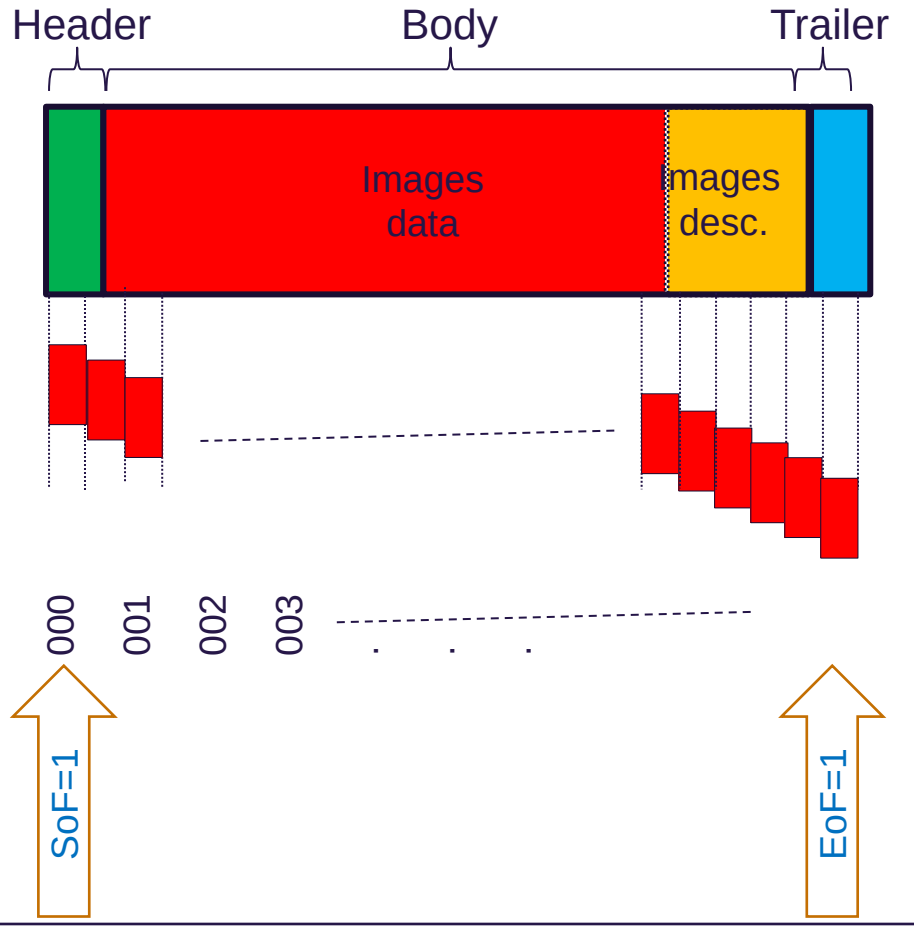
- Two applications are under development
  - Train builder emulator
  - PCL node application
- Data-driven model (push)
  - Start PC nodes
    - ➔ Wait for data
  - Start UDP feeder
    - ➔ Wait for timer signal
  - Start timing system
- Communication model
  - 1-to-1, N-to-N



TB data format

- Train structure (configurable)
  - Header, images & descriptors, detector data, trailer
- Implementation
  - Plain vanilla UDP
  - The train is divided into chunks of 8k (configurable)
- Packet structure





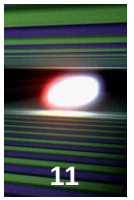
European XFEL

Tuning summary I

- Driver
  - Newer driver with NAPI enabled - ixgbe-3.9.15
  - InterruptThrottleRate=0,0 (=disabled, default 16000, 16000)
  - LRO (Large Receive Offload) =1 (default)
  - DCA (Direct Cache Access)=1(default)
- Linux kernel (Socket buffers)
  - net.core.rmem_max=1342177280 (default 131071)
  - net.core.wmem_max=516777216 (default 131071)
  - net.core.rmem_default=10000000 (default 129024)
  - net.core.wmem_default=10000000 (default 129024)
  - net.core.netdev_max_backlog=416384 (default 1000)
  - net.core.optmem_max= 524288 (default 20480)
  - net.ipv4.udp_mem = 11416320 15221760 22832640 (default 262144 327680 393216)
  - net.core.netdev_budget = 1024 (replaces max_backlog for NAPI driver)

European XFEL

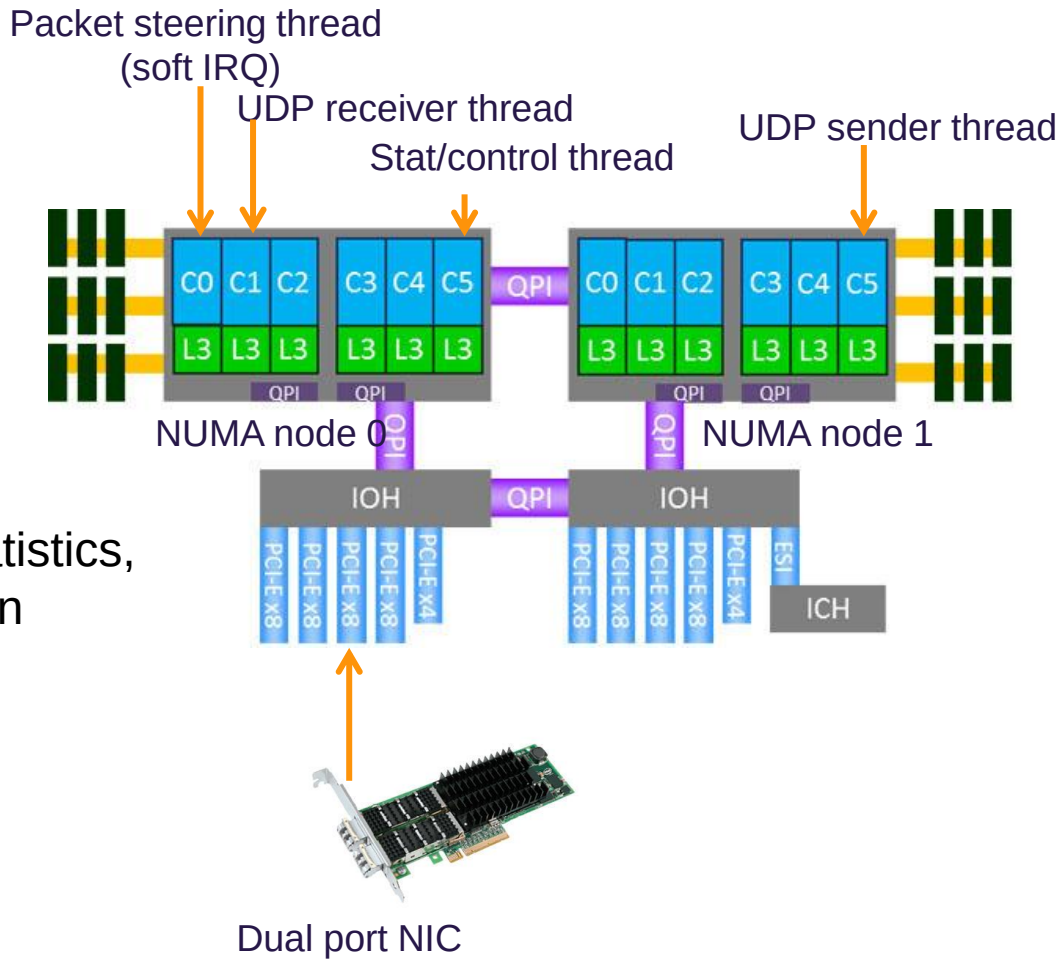
Tuning summary II



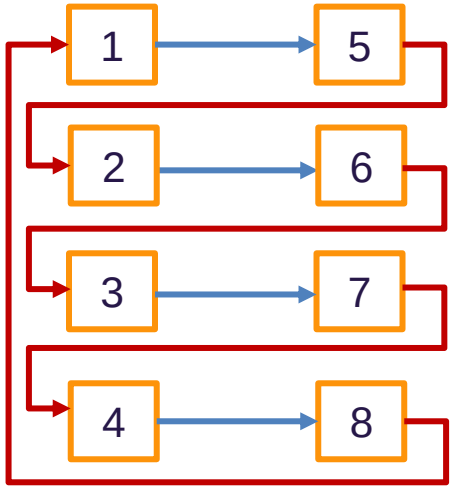
- NIC:
  - Jumbo frame: MTU=9000
  - Ring buffer RX & TX= 4096 (default 512)
  - rx-usec= 1 (default 1)
  - tx-frames-irq= 4096 (default 512)
- PCI bus
  - MMBRC (max memory byte read count) = 4096 bytes (default 512)
- Linux services
  - Services iptables, irqbalance, cpuspeed are stoped
- IRQ affinity: All Interrupts are handled by Core 0
- Hyperthreading: no visible effect (ON/OFF)

Test setup

- Application setting
  - Core 0 is exclusively used for processing IRQ
  - Sender thread on core 7
  - Receiver thread on core 2 (even number, except 0)
  - Some processing is done on received packets
  - Separate thread to compile statistics, write to log file run in parallel on core 5 (rate every 1sec)
  - Run in root mode: necessary to set scheduling policy and thread priority



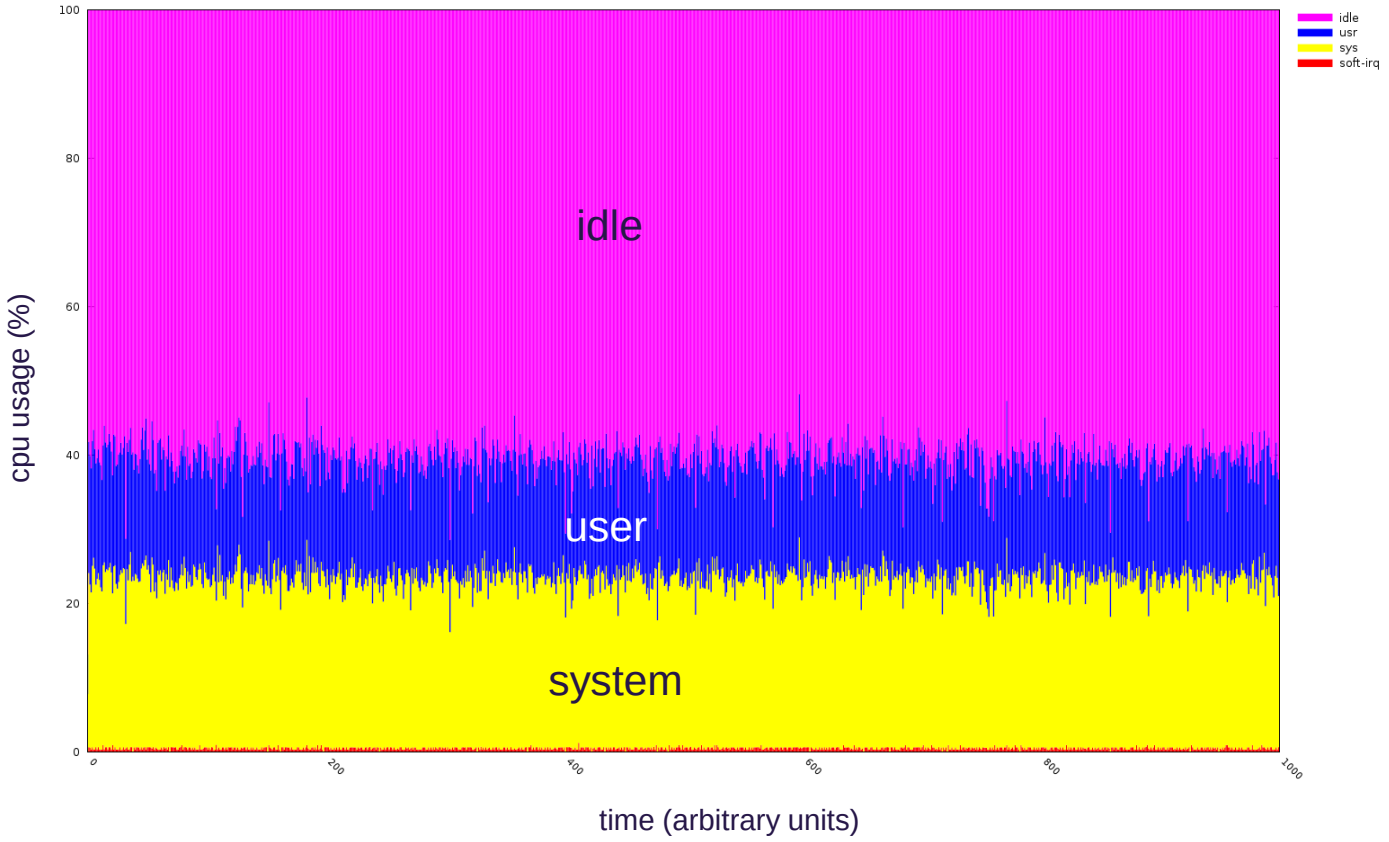
Test results



- Pairs of sender-receiver run in parallel on all machines
- UDP packet size 8192, configurable
- Run length (#packets/total time):
  - 10^6 (12.21 sec, 6.6 seconds without time profile)
  - 10^7 ~ 2.5×10^9 (several times w/out time profile)
  - 5×10^9 (16h37m)
- Emulating realistic timing profile from TB
  - Each node sends 1 train (~131030 packets), then wait for 0.7 sec
- Packet loss
  - Few 10s~100s packets might be lost
  - When? Loss is observed only in the beginning of a run (warm-up phase?)
  - Packet loss rate $<10^{-9}$ (excluding warm-up phase)
  - Where? All machines, few machines, no machine

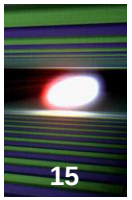
CPU usage

- UDP reader thread (core 2)



European XFEL

Summary



- Tuning is required to achieve full bandwidth for 10GE
- Additional kernel, driver and system and application tuning required to reduce UDP packet loss