LS1/LS2 System Architecture Changes

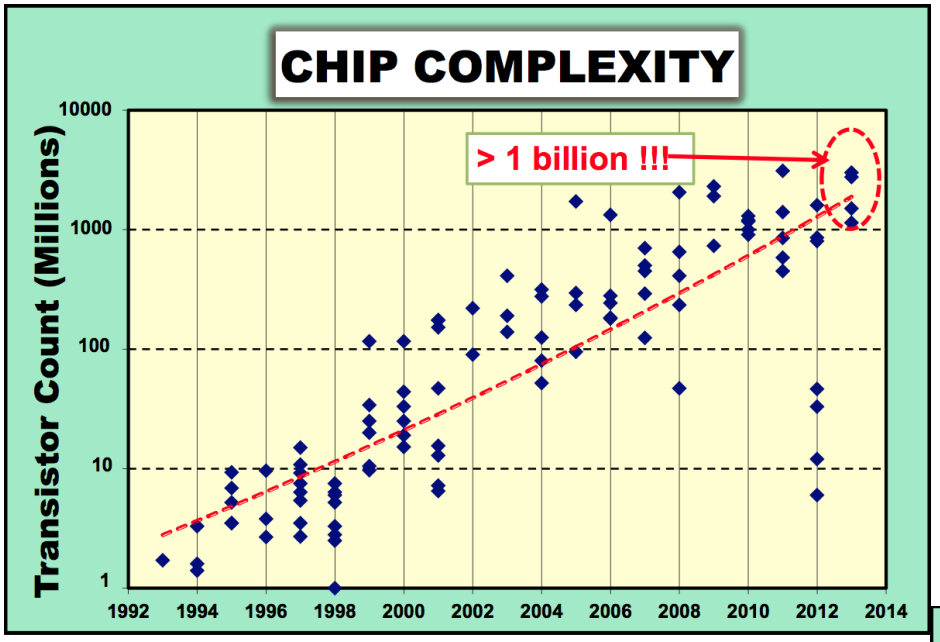
Marc Dobson (CMS)
On behalf of ALICE, ATLAS, CMS and LHCb

Content

- Crystal ball gazing on the technology impact for future system architectures
- Trends and outlook:
  - Moore's Law for processors
  - Multi-multi-core processors, GPGPU, co-processors (Intel Xeon Phi), APUs
  - Networking to the motherboard of 10/40/100 GbE, Infiniband, or new technologies, network virtualization
  - OS layer, Virtualization, cloud
  - PC HW, blades, highly compact servers, micro-servers
  - NAS, SAN, cluster files systems, NFSv4
  - Application management
- Current Usage of this Technology
- Future architectures: in what ways can this technology be used in our environments and what impact does it have on our system architectures
  - Future L1 Trigger & FE Links
  - Future Readout & Readout Link
  - Single large Network for control and data
  - Single Large Farm
  - Single Large File System

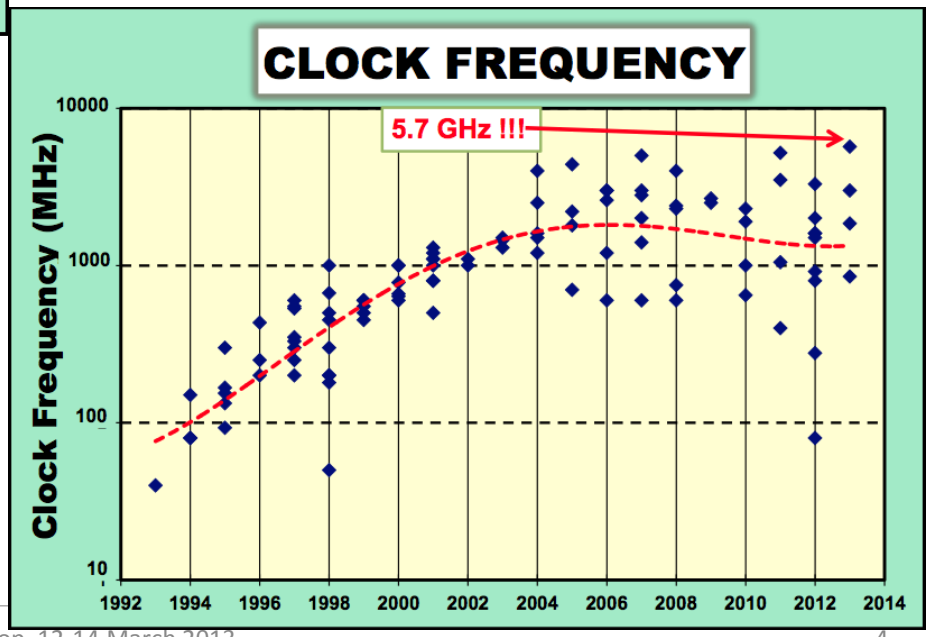
Trends and Outlook

CPU Complexity & Frequency Trends



The complexity increase of processors (number of transistors) is still following Moore's Law (doubling every 18 months)

Source: 2013 International Solid-State Circuits Conference (ISSCC) trend report

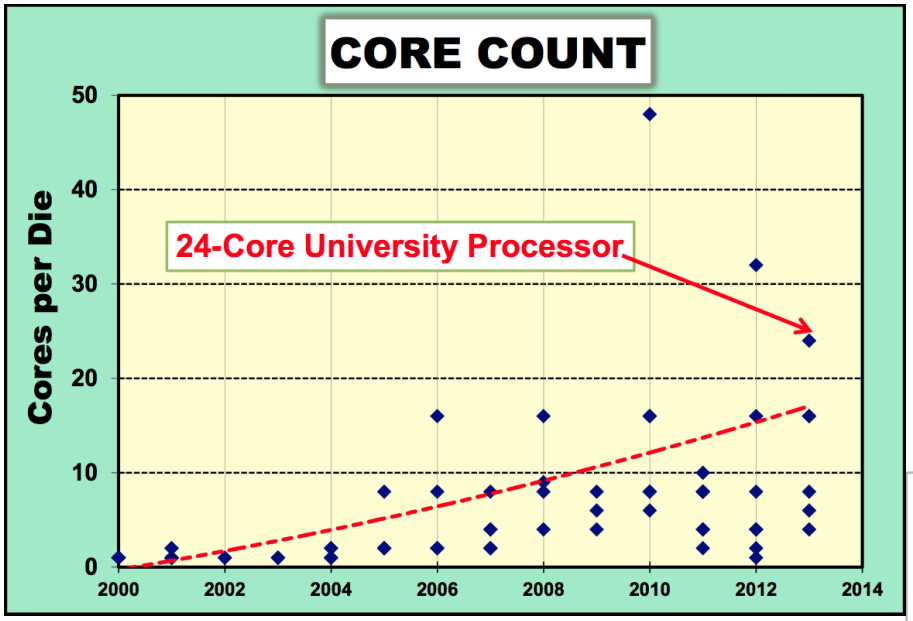


The frequency of processors has been leveling off since about 8 years

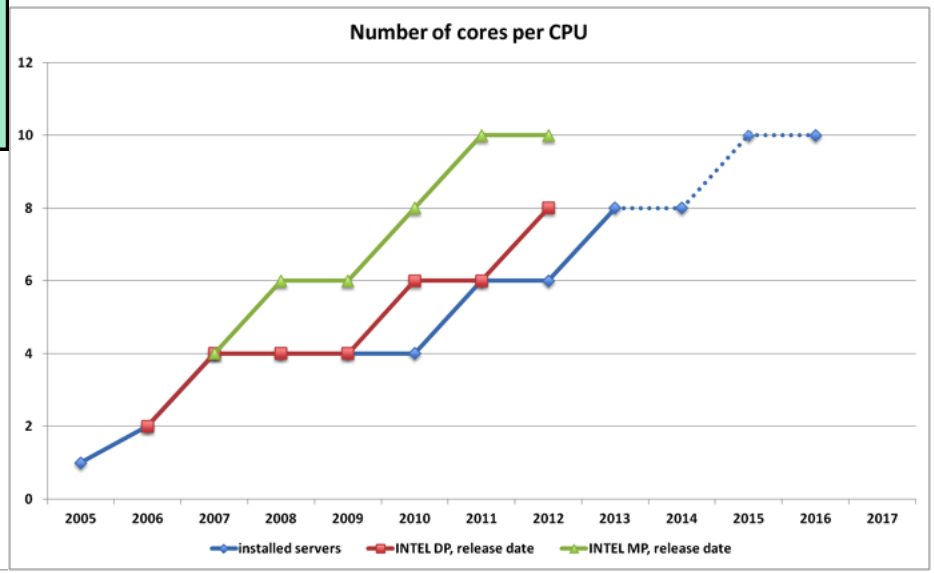
Source: 2013 International Solid-State Circuits Conference (ISSCC) trend report

Multi-multi-cores

- The trend is in more and more cores per device



Core count per Die, from the 2013 ISSCC trend report. Shows a nearly linear progression of 1-2 cores/year.



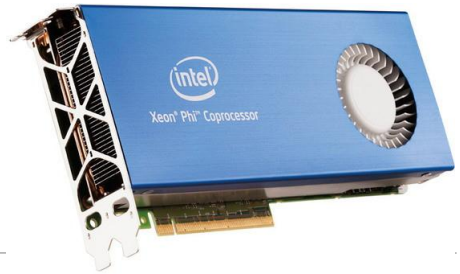
Core count per processor, for CERN IT installed servers, Intel Dual Socket, or Intel Multi Socket (reference is physical cores)

GPU versus co-processors

- Trend in using GPU for certain computation
  - Pure computation is impressive, however needs …
  - Specific development tools (specialized manpower)
  - Refactoring of code (time consuming & expensive)
  - Overhead of getting data in/out of the device
  - Need stripped access on large data sets

	# Cores	# Transistors [Billion]	SP GFlops	DP GFlops	Structure Size [nm]
Nvidia Tesla K20X	2688	7.1	3951	1317	28
AMD FirePro S10000	3584	8.62	5910	1480	28
Intel Xeon Phi SE10X	61	?	2140	1070	22

- Has industry heard us? Are the co-processors back?
  - Intel had come up with the Xeon Phi co-processor with a simplified X86 instruction set, which can run Linux natively
    - Refactoring of code should be smaller (less time)
    - Use of more standard tools (gains probably less also)
    - Need to be able to use highly vectorized computation

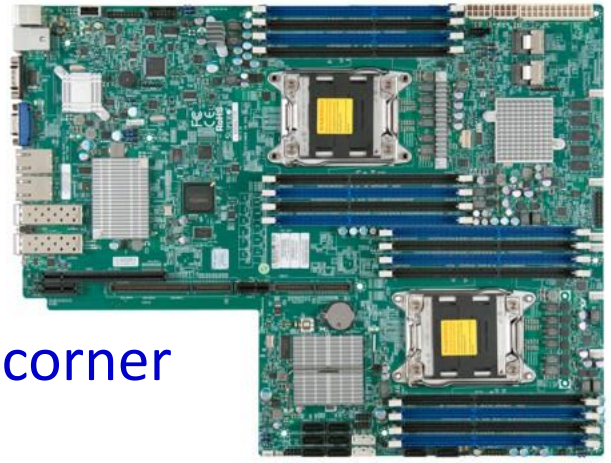


What else?

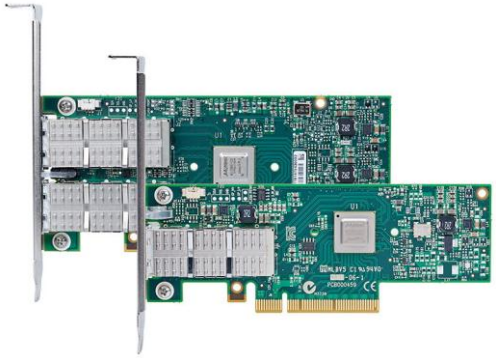
- APUs or Accelerated Processing Unit
  - Coined by AMD, with its AMD Fusion technology
    - Appeared in 2011
    - In 2012: Trinity with 4 Bulldozer cores + 128 to 384 Stream Processors (GPU)
  - Pushing this as a standard with the name Heterogeneous System Architecture (HSA)
  - Basically low power, low-ish core count, integration eases communication, but still requires code to be re-factored
  - Off-chip communication is the slowest link, i.e. memory
  - Good for mobile devices. How about DAQ?
- ARM CPUs
  - Low power RISC processors
  - Mobile market
  - Up to 4 cores

Future of networking

- (1+)x 10GbE ports on the motherboards today
  - Can the BW be used?
  - Separate 1Gb control network?
    - What about IPMI?



- 40GbE/Infiniband on MB is just around the corner
  - Widely available as PCIe cards or mezzanines
  - Same questions apply?



- What about 100GbE onboard?
- Future networking interfaces integrated to the processor?
  - Higher BW/faster Links/lower latency to the CPU

Virtualization

- Virtualization is all the rage
  - Most useful for easily moving heterogeneous applications from one physical system to another (especially if tied to OS)
- Can DAQ use it? Is DAQ using it to its full potential?
  - Where can it help?
  - DAQ run control SW is already capable of starting apps wherever one needs, wants (more or less), no strong OS dependence, virtualization overhead to take into account
  - Used to take advantage of the multi-core devices available today with applications only needing small numbers of cores, or wanting/needing independence with respect to other processes
  - Offline cloud usage
  - Other DAQ services? Can help with irreducible single points of failure.
- What about SysAdmin services?
  - Used to virtualize classical services (DNS, DHCP, HTTPD, LDAP)
  - Can everything be virtualized? Should it be?
    - Probably useful for some things: NX servers, boundary nodes (keep state, and have user independence)

HW: micro/cloud/standard servers, blades

- Micro servers (low power, low # core)
  - Popular for “cloud” like clusters
  - Cheap, single app
  - Well suited to HLT
    - Non-competitive if HLT becomes thread friendly and saves on memory-per-core (RAM costs)

SeaMicro, 10U, 64 quad-core Intel Xeon E3-1260L

- Cloud servers (compact multi-core machines)
  - Extensively used for HLT
  - Cheaper than 1U servers, save on space and power

- Standard servers:
  - Good for specific functions, if specialized interfaces needed, or low performance throw away HW
- Blades servers:
  - Very nice from a management point of view, compactness, high performance, ideal for virtualization
  - Used for specific services: DCS, SysAdmin, DAQ, Online DB

NAS, SAN, Cluster FS

- Are we thinking big enough?
- Classic SANs are often disappearing in favor of NAS integrated in the global network
  - Your SAN is your network
- Cluster File Systems are all the rage
  - Do they work outside the lab?
  - What are the benefits?
  - Isn't everything a file, somewhere?
  - Can it leverage the many large disks on a cluster?
  - Redundancy is built-in: how does it work in reality if a complete rack disappears? What about monitoring? Control on allocation algorithm?
  - Impact of cluster redundancy on the network? Separate "heartbeat" network? Bonded links for redundancy?

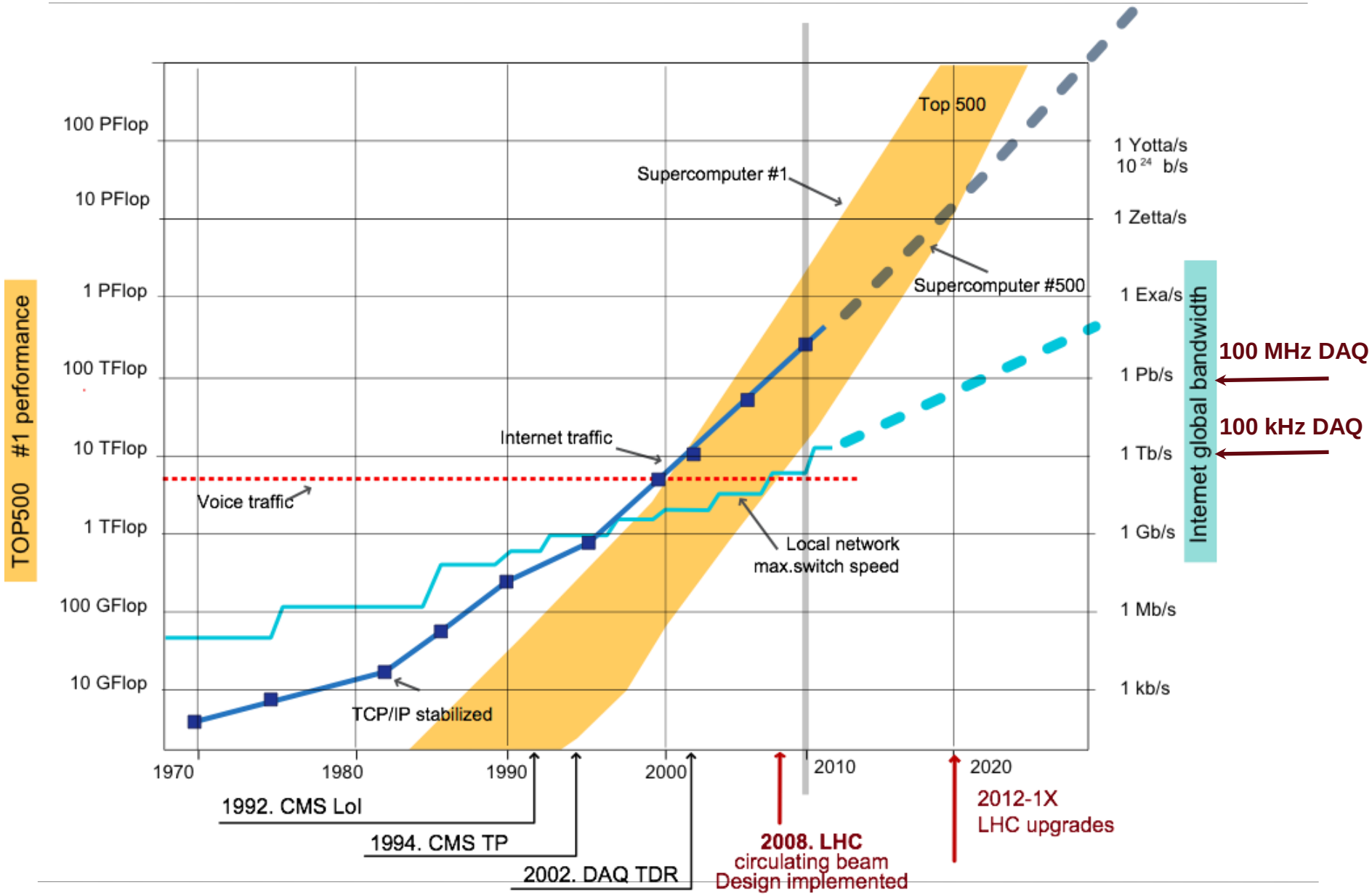
OS Layer vs Hypervisor Layer

- Will the OS become a thinner layer over hypervisor?
  - Same kind of evolution as micro-kernels (everything runs as services)
- Hypervisor has 4-5% overhead
  - More and more HW support for virtualization, e.g. newest Intel network card has it, but how can it be used?
- What is critical to our (data taking, HLT) performance?
  - Disk IO usually isn't (exceptions of the temporary online storage)
  - Network is
    - What is the current overhead?
    - Is anything being done to decrease this?

Application or Infrastructure Management

- CERN used to be a “Big Fish”, now there are bigger fish in the sea: Google, Yahoo, Facebook,…
- More cluster software generally available, evolving quickly
  - Keep our eyes open to find and use it (minimize maintenance)
  - OpenStack, etc…
- CERN IT are now following the trend
  - “Agile” infrastructure
- DevOps perspective to application development, deployment & system administration
  - Teams are more integrated and work closely together to improve the overall running system
    - Developers work with SysAdmins to understand the impact and OS level solutions
    - SysAdmins work with developers to understand their requirements and constraints

Outlook Summary



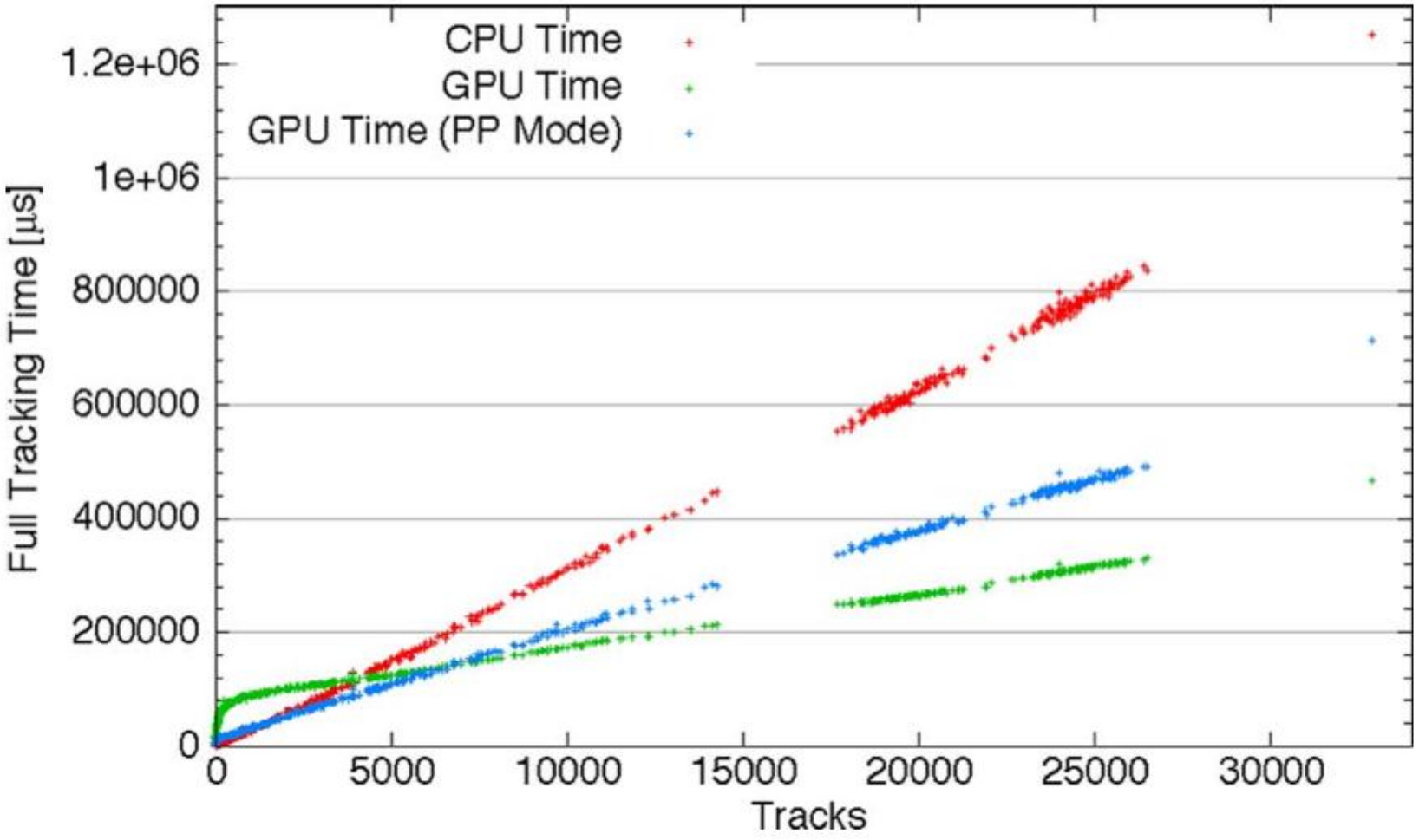
Current Usage of this Technology

The reality of DAQ or services today, I

- Are multi-cores used efficiently?
  - Not from a memory point of view: just N processes running on N cores with no sharing of memory (maybe some caching effects)
  - Some applications are multi-threaded and can use the number of cores, but very few (in percentage of machines, where HLT dominates)
  - The memory is just multiplied up (N * 2GB)
- Are GPUs or co-processors used?
  - Alice: HLT TPC tracker algorithm ported to GPUs & used for 2010/2011 Pb-Pb runs (9 months to rewrite, 3 times faster than CPU)
  - ATLAS: initial studies for HL tracking algorithms
  - Apart from Alice, nothing currently used online or offline
  - Need to come from offline or integrated in offline as this is used for HLT
  - Issues with latency of getting data in/out and having enough parallelism and data to make it worthwhile
  - Refactoring is time consuming especially if vectorizing and streaming is limited
    - Maybe more gains from just rethinking the way the code is written (view frameworks and data structure in a better way [read computer way])

ALICE Tracker GPU Acceleration

CPU/GPU performance for different event sizes



The reality of DAQ or services today, II

- Is 10GbE used currently?
  - Data networks in specific areas maybe
  - Online Databases (Oracle)
  - Even in future the link is likely to be under-used (cannot sustain the processing for this BW)
- What about higher rates?
  - In specific areas after LS1, minimal processing, mainly data stream merging, feeding of HLT farmlets
    - ROS in ATLAS, RU/BU in CMS
  - After LS2 more common:
    - Widespread in LHCb from Readout to CPUs

The reality of DAQ or services today, III

- Virtualization is being used in isolated areas by some exp.
  - Icinga/Nagios servers, gateways, public nodes, infrastructure services, Quattor/Puppet servers
- For after LS1 plans are:
  - Use for more SysAdmin services (not bootstrap services)
  - Use for DAQ services:
    - Run control services
    - Monitoring services
    - Sub-detector services, local event building and analysis
    - DCS (ATLAS getting rid of isolated HW, & LHCb)
- What about a “virtual Data Acquisition System”
  - Not there yet due to specific network or HW constraints
  - ALICE looking at it for the Event Builders: maximize use of available HW

Industry balance: CERN affected?

- What is the industry balance?
  - Lower the TCO (total cost of ownership)
    - Reduction in power consumption: power efficiency, DC feeds to racks
    - Reduction in cooling required: free air cooling, no rack ventilation
    - Optimize usage of nodes (pool resources)
- And CERN?
  - Traditionally outside IT, different people pay for those different areas, therefore no overall plans
  - Changing where possible (infrastructure already existing)
    - Power efficiency, PC costs, optimizing usage (making use of multi-cores, multiplexing the usage)
    - Blades or micro-servers (SeaMicro “fabric”)
    - Open Compute?

Impact of Technology on future Architectures

What could a future DAQ look like

if it overcame all the “if”s and “but”s, and “maybe”s?

Future L1 Trigger & FE Links

- What about a L1 rate of 500kHz, 1MHz or even 40MHz?
  - Limited by latency and BW (except LHCb)
    - At least with existing detector FE electronics (3-6.7us for CMS, 20us for ATLAS)
  - Limited by algorithms in L1 Trigger
    - Multi level with tracking at L1
- Start from scratch
  - New detector links/electronics
    - Requires High BW rad-hard link & low power electronics (power dissipation on detector)
    - For example v2 GBT link (see talk by Jorgen Christiansen), but worried it is already too late for some development
    - Could have longer pipeline buffers on FE (increased L1 latency)
    - Need longer buffers at the DAQ Readout: not such an issue
  - Force detectors to do better: DAQ usually not the bottleneck
    - LHCb design for after LS2

Future Readout & Readout Link

- Readout or off-detector electronics
  - Higher BW, bigger buffers
  - Basically DCS interface to FE & FE Link to DAQ converter (maybe some data processing or formatting)
  - Potential merging of FE Links
- Readout Link:
  - Standardize to commercial HW and Industry standard protocols, e.g. 10/40GbE, TCP/IP/Infiniband
  - Typically envisaged for CMS FEROL link, ATLAS ROS, LHCb Tell40
- This has all the features of the Tell40 in some format!

Single large Network

- Go directly to a single phase event building
  - Care has to be taken for head of line blocking or source level buffering
  - Issue with streaming from custom electronics to many source (O1000)

- Single large network
  - 1152 * 10GbE ports available today (Huawei)
  - PCs available now with 10GbE on-board
  - Do away with control network (what about IPMI???)
  - Implement QoS, VLANs or virtual networking
    - Separate data, control traffic
  - See the LHCb LS2 plans
  - See the next talk by Niko Neufeld

CE12812
Huawei

Single Large Farm

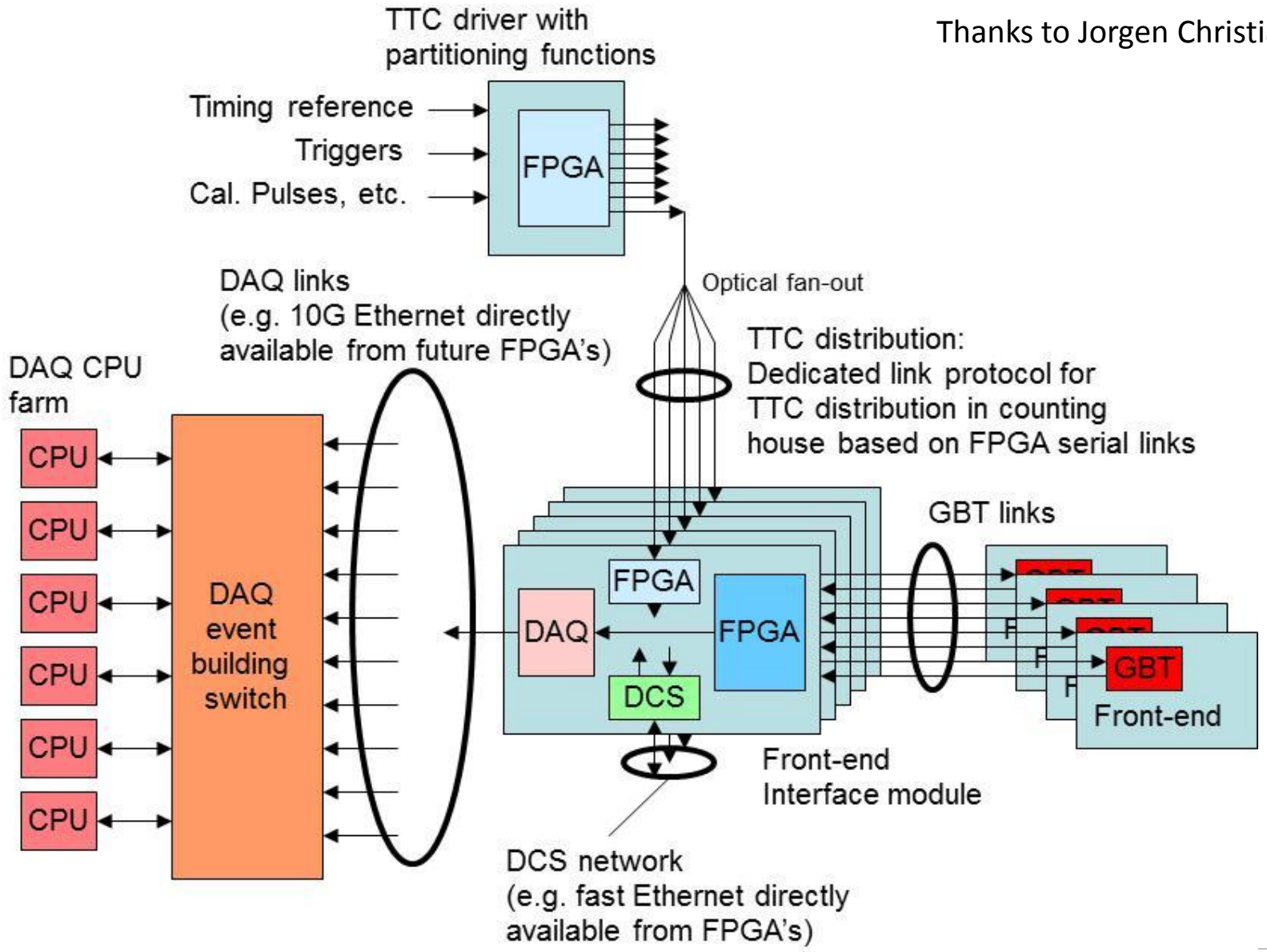
- Event Building and HLT in the same nodes
  - Multi-cores are here, multi-multi-cores are round the corner as well as co-processors
    - Largest part of system is HLT or data analysis which is a high CPU consumer
  - Need to match network BW and processing power
    - Careful balance, but 10GbE and next generation processors should be a reasonable match
    - Can 40GbE be useful with co-processors or more cores? Probably
- DAQ services
  - Just a question of accounting: i.e. where is what running and even then do we need to know?
  - Any application running anywhere, full connectivity
  - Could run all DAQ services as virtual machines or not
  - Already running offline as a Cloud (do not care where it runs, nodes just advertise they are up and ready)
  - Could run the detector services anywhere (virtual or not)
    - Do we care where the VME crate control or uTCA crate control is running? NO
  - The only exception is for attached HW which is disappearing
    - Becoming network attached HW (USB to Ethernet bridges, uTCA & TCA Ethernet communication)
- SysAdmin Services
  - Most are run anywhere services (exceptions are periphery/boundary nodes or HW attached)
  - Also most service could be virtualized. Only a few exceptions: bootstrap servers
  - Gives redundancy and reliability

Single Large File System

- Cluster File system over the entire cluster
  - Redundancy and high availability, robust against failures (distributed DDP)
  - Use available disk space
- What are the possible uses?
  - Replace central file servers (NAS) ?
    - Probably not completely
  - Replace event buffer storage?
    - Being looked at in CMS
  - Caching of events for later processing: “parked data”
- What is the impact on network usage? (distribution of the data across the cluster)
- What about more classical File Systems?
  - pNFS (no server support in Linux yet), available on NAS systems usually.

What would it look like ?

Thanks to Jorgen Christiansen



Conclusion

- Many interesting developments and paths forward
  - What is feasible, on what timescale, with what benefits and what costs?
  - Identify clear benefits and feasible tasks
  - Identify and privilege cross experiment developments
  - Define impact/requirements on other systems, e.g. sub-detectors and their designs, upgrades
  - Do not forget why we are doing this: Physics
- The scene of DAQ can be radically different in the future
  - Are we looking forward and embracing it?
  - Or are we going to stay with tried and tested methods

I think the former is more the “CERN spirit”

Thanks

- David Francis (ATLAS)
- Christoph Schwick (CMS)
- Sergio Ballestrero (ATLAS)
- Bernd Panzer-Steindel (IT)
- Cristian Contescu (ATLAS)
- Niko Neufeld (LHCb)
- Sergio Cittolin (CMS)

- And all speakers which allowed me to refine ideas