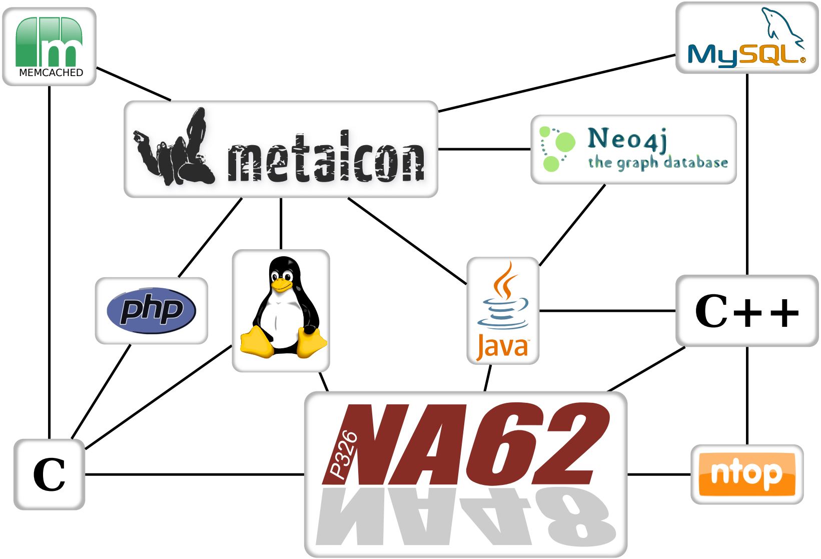
Jonas Kunze: IT affine physicist



How to analyze 12 GBps data online

Jonas Kunze

University of Mainz

29.03.2012

SPONSORED BY THE

Federal Ministry
of Education
and Research

1. Motivation
   - The physics
   - The experiment
2. Trigger topology
   - Three level online trigger
   - New proposal
3. Implementation
   - Socket programming
   - Parallel programming
4. Conclusion

1 Motivation
- The physics
- The experiment

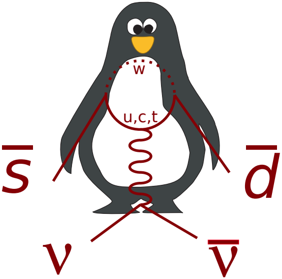
2 Trigger topology
- Three level online trigger
- New proposal

3 Implementation
- Socket programming
- Parallel programming

4 Conclusion

The Kaon K^+

- An elementary particle like Proton, Neutron...
- Decays within 12 ns

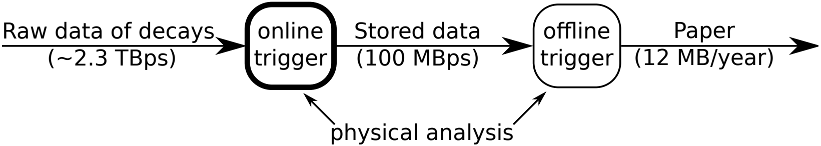


$K^+ \to \pi^+ \nu \bar{\nu}$

Extremely rare: 1 out of 10^{10} decays

What it does mean

1 out of 10^{10} decays is like taking a 227 kB picture of every human being on earth and filtering those with red hair, green-brown eyes, 3 birthmarks at the right cheek, being smaller than 1.50 m . . .



Raw data of decays (~2.3 TBps)

online trigger

Stored data (100 MBps)

offline trigger

Paper (12 MB/year)

physical analysis

The NA62 Experiment at CERN

- About 90 physicists from I, GB, RU, BE, USA, DE . . .
- Will measure about 100 decays within 2 years
- Needs to analyze 10^{13} K^+ decays

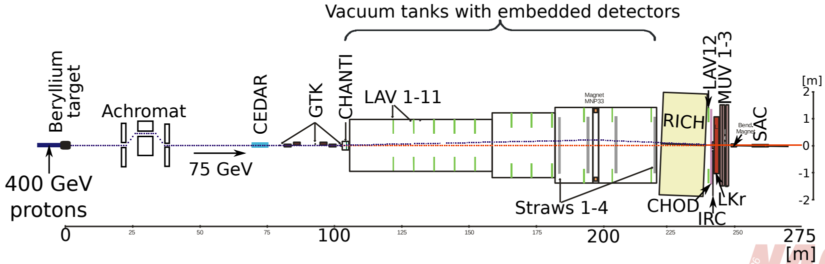
NA62 Experiment at CERN

K^+ production

High energy protons colliding with a beryllium target

Measurement

- 0.8 GHz particles crossing
- 9 “cameras” shooting a picture at every decay (10 MHz)



Data rates

10 MHz event rate or “10 Mio. pictures per second”

Detector	Event size [B]	Data rate [GBps]
CEDAR	216	2.16
GTK	2250	22.50
CHANTI	192	1.92
LAV	160	1.60
STRAW	768	7.68
RICH	160	1.60
MUV	768	7.68
IRC & SAC	576	5.76
LKr	**222 k**	**2220**
Sum	**≈227 kB**	**≈2.3 TBps**

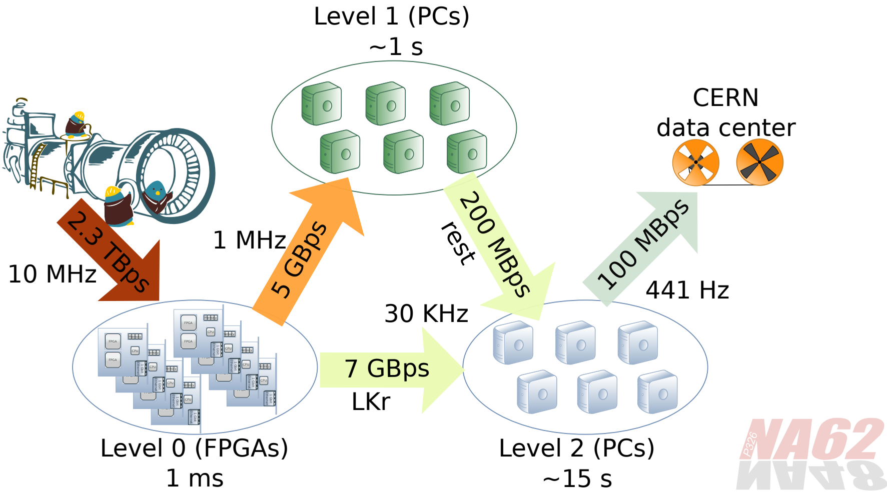
It’s like 10 Mio. users uploading a big profile image every second!

1. Motivation
   - The physics
   - The experiment
2. Trigger topology
   - Three level online trigger
   - New proposal
3. Implementation
   - Socket programming
   - Parallel programming
4. Conclusion

Online trigger system

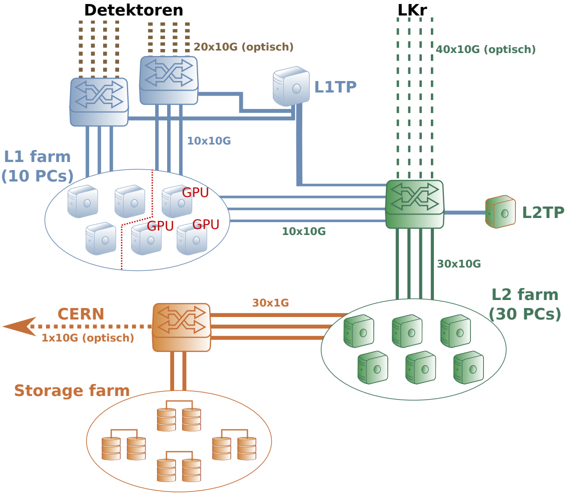
Three levels to filter data

Data transmission via ordinary 10 gigabit ethernet and **UDP/IP**:



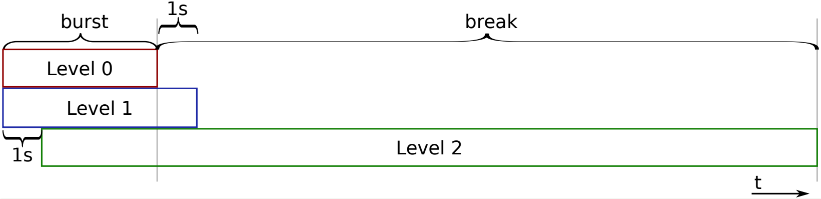
First topology proposal

Original concept:



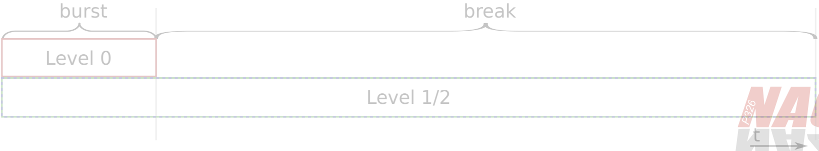
Burst time

Only 3-9 sec. proton burst and long break



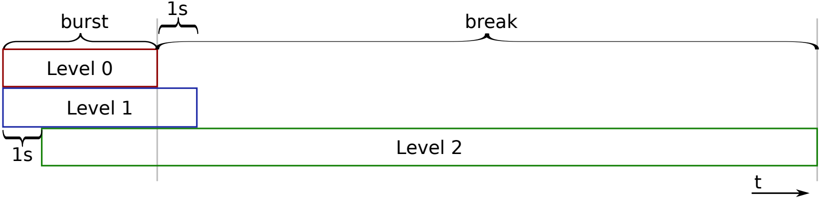
My proposal to use resources more efficiently

Combine L1 and L2 to one farm



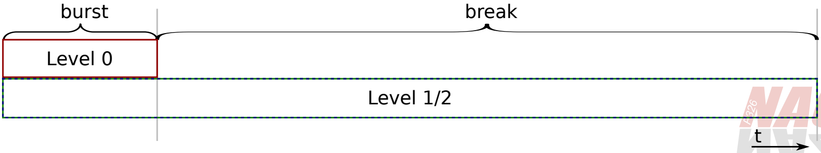
Burst time

Only 3-9 sec. proton burst and long break

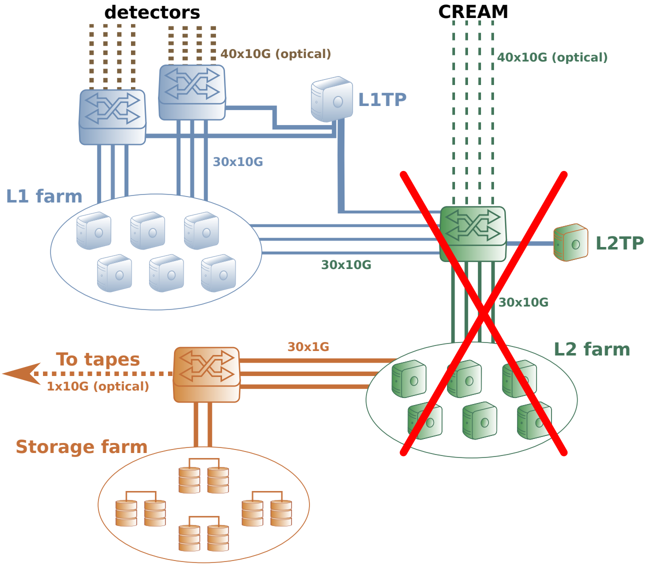


My proposal to use resources more efficiently

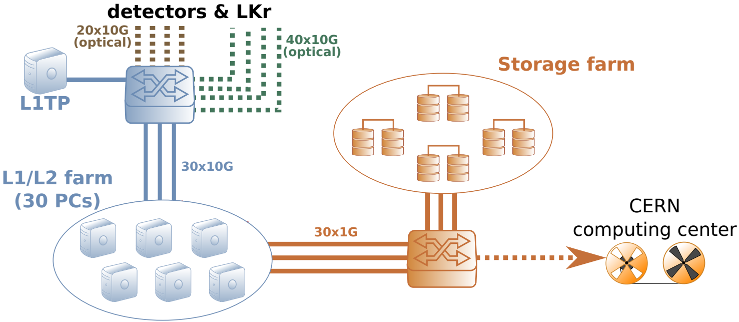
Combine L1 and L2 to one farm



Don't separate L1 and L2!



Combine L1 and L2 to one farm

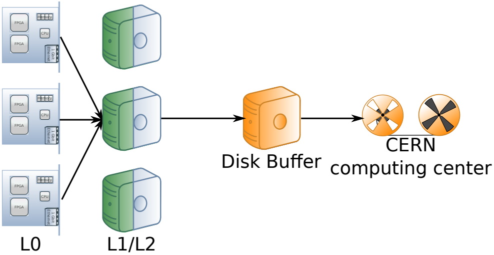


We save about 80k

- No L1 PCs anymore
- Less switches, less network cards

New proposal

Event building @ L1



Every subdetector sends data of one event to the same PC

- ✚ **More physics at earlier state**
- ✚ No broadcast of a L1 decision needed anymore
- ✚ Easier to implement load balancing (self-sustaining PCs)

1. Motivation
   - The physics
   - The experiment
2. Trigger topology
   - Three level online trigger
   - New proposal
3. Implementation
   - Socket programming
   - Parallel programming
4. Conclusion

Bad performance with interrupts

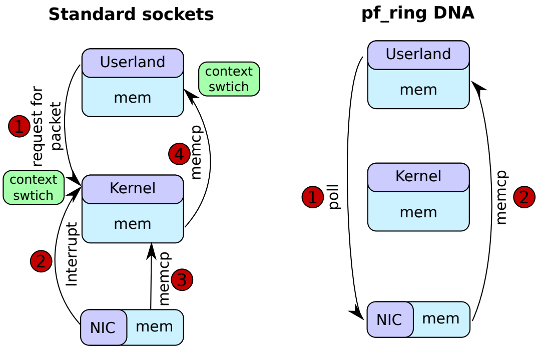
Standard socket programming is interrupt based

Every hardware and software (syscall) interrupt induces a context switch ($\approx$100 ns)

⇨ High packet loss ($> 10^{-5} \Rightarrow$ loose $>$100 Mio. events)

⇨ No problem for web apps but I cannot use kernel sockets!

Solution: pf_ring DNA - new type of network socket



pf_ring DNA

Special socket: pf_ring DNA by ntop (open source)

- Polling the NIC memory directly (avoids system calls)
- Only ≈40% CPU @ full speed 10 G receiving 1 kB packets
- No packet loss at all

pf_ring does not yet support any protocol

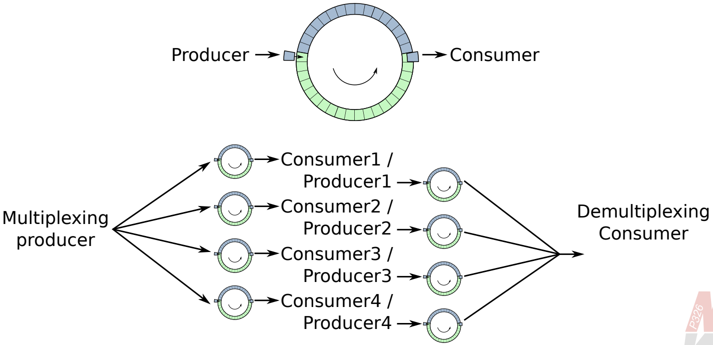
- Every byte has to be moved by the user
- I had to implement Ethernet, IP, UDP, ARP and IGMP

⇨ 270 kHz Eventbuilding rate with ¡5 virtual cores
⇒ ¿19 cores left for L1 and L2 trigger

Same problem with Mutex/Semaphore

Bad performance with Mutexes/Semaphores

Implemented lockless queues based on consumer-producer communications



Producer

Consumer

Multiplexing producer

Consumer1 / Producer1

Consumer2 / Producer2

Consumer3 / Producer3

Consumer4 / Producer4

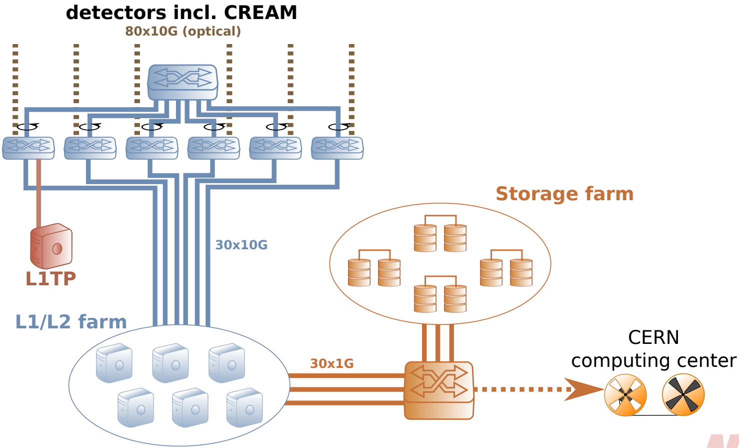
Demultiplexing Consumer

1. Motivation
   - The physics
   - The experiment
2. Trigger topology
   - Three level online trigger
   - New proposal
3. Implementation
   - Socket programming
   - Parallel programming
4. Conclusion

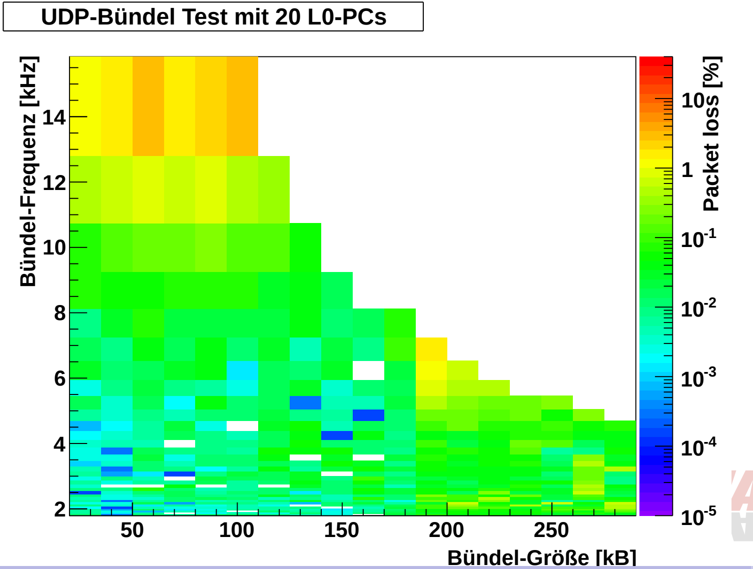
- High energy physics $\hat{=}$ high data rate
- A well planned strategy can save a lot of money
- Using ordinary 10G ethernet saves money but lossless communication only feasible with special software: pf_ring DNA
- High performance parallel programming requires special approaches

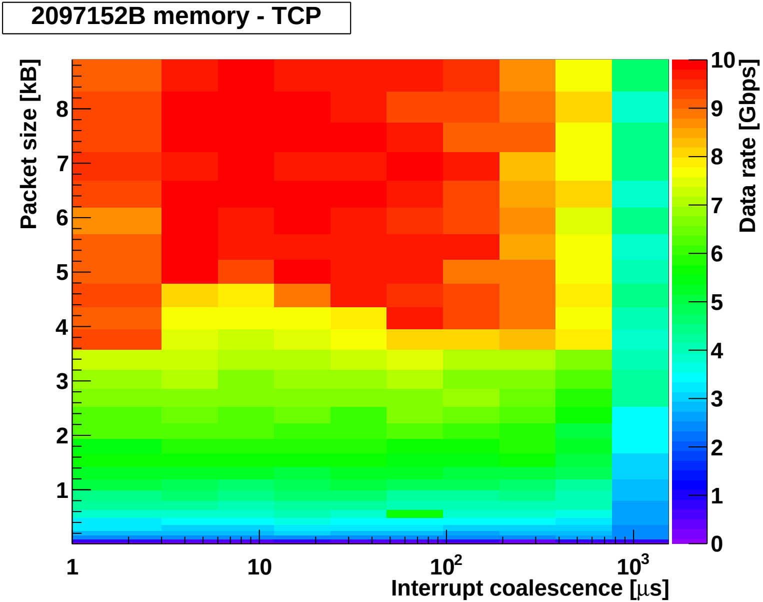
Thank you for the invitation!

Tree topology (Hexapus)

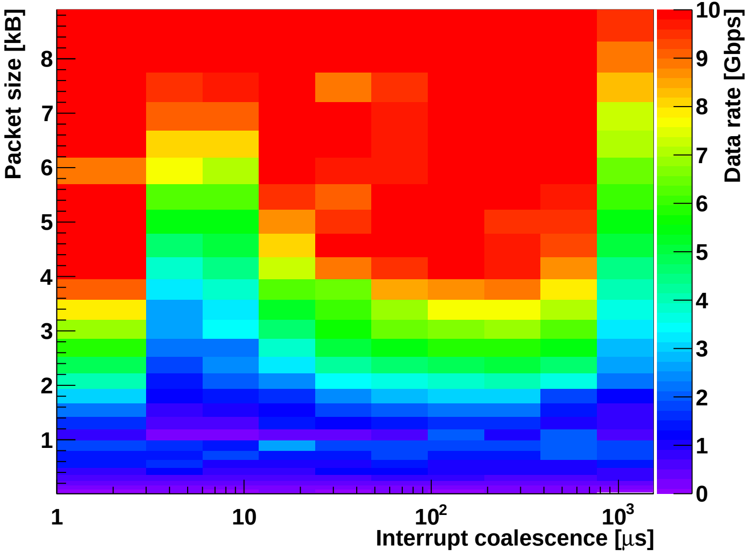


Packet loss


UDP-Bündel Test mit 20 L0-PCs
Bündel-Frequenz [kHz]
2
4
6
8
10
12
14
50
100
150
200
250
Bündel-Größe [kB]
Packet loss [%]
10
1
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}


2097152B memory - TCP
Packet size [kB]
1
2
3
4
5
6
7
8
1
10
10^2
10^3
Interrupt coalescence [µs]
Data rate [Gbps]
0
1
2
3
4
5
6
7
8
9
10

2097152B memory - UDP


Packet size [kB]
10^2
10^3
Interrupt coalescence [µs]
Data rate [Gbps]