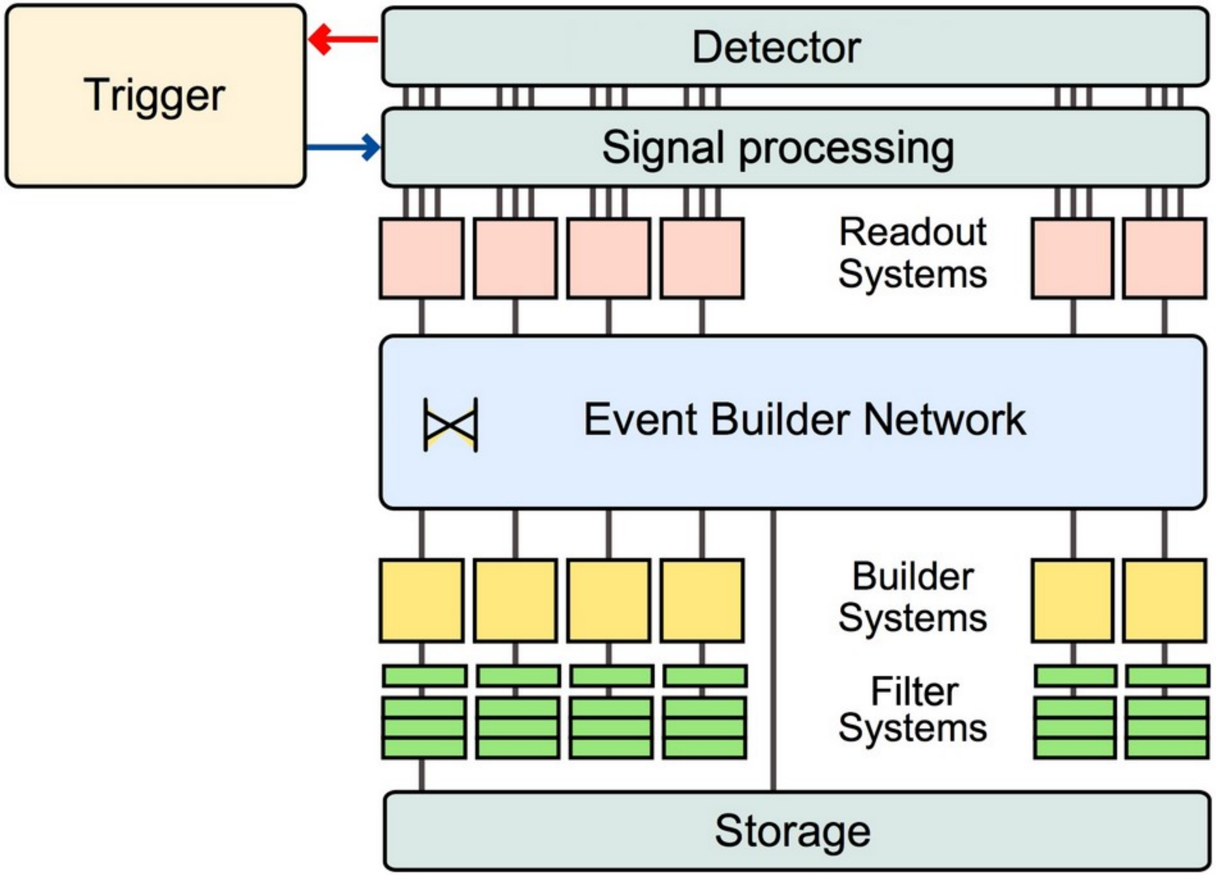
DATA ACQUISITION
ELECTRONICS & TRIGGER

Tommaso Colombo
CERN

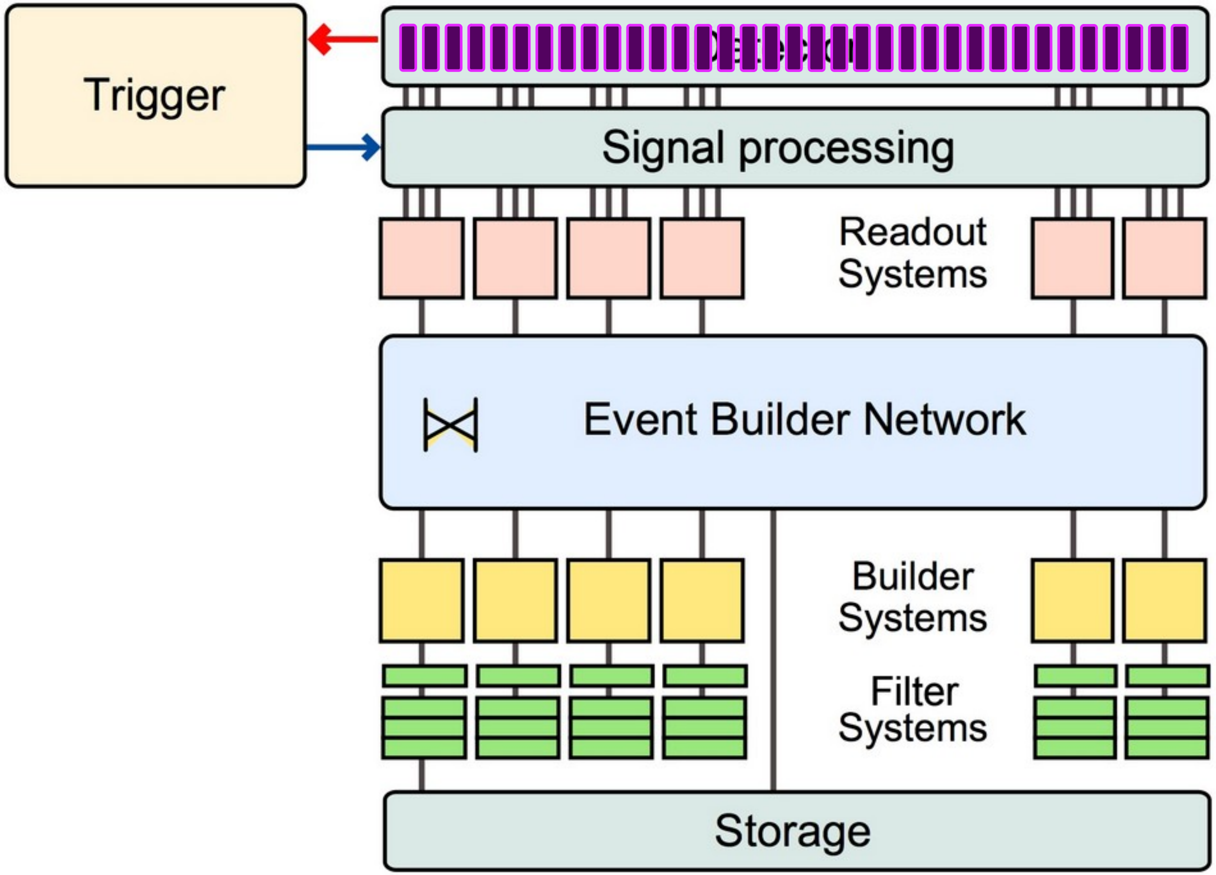
Summer Student Lectures Programme
CERN, 24 July 2024

DATA COLLECTION
OVERVIEW

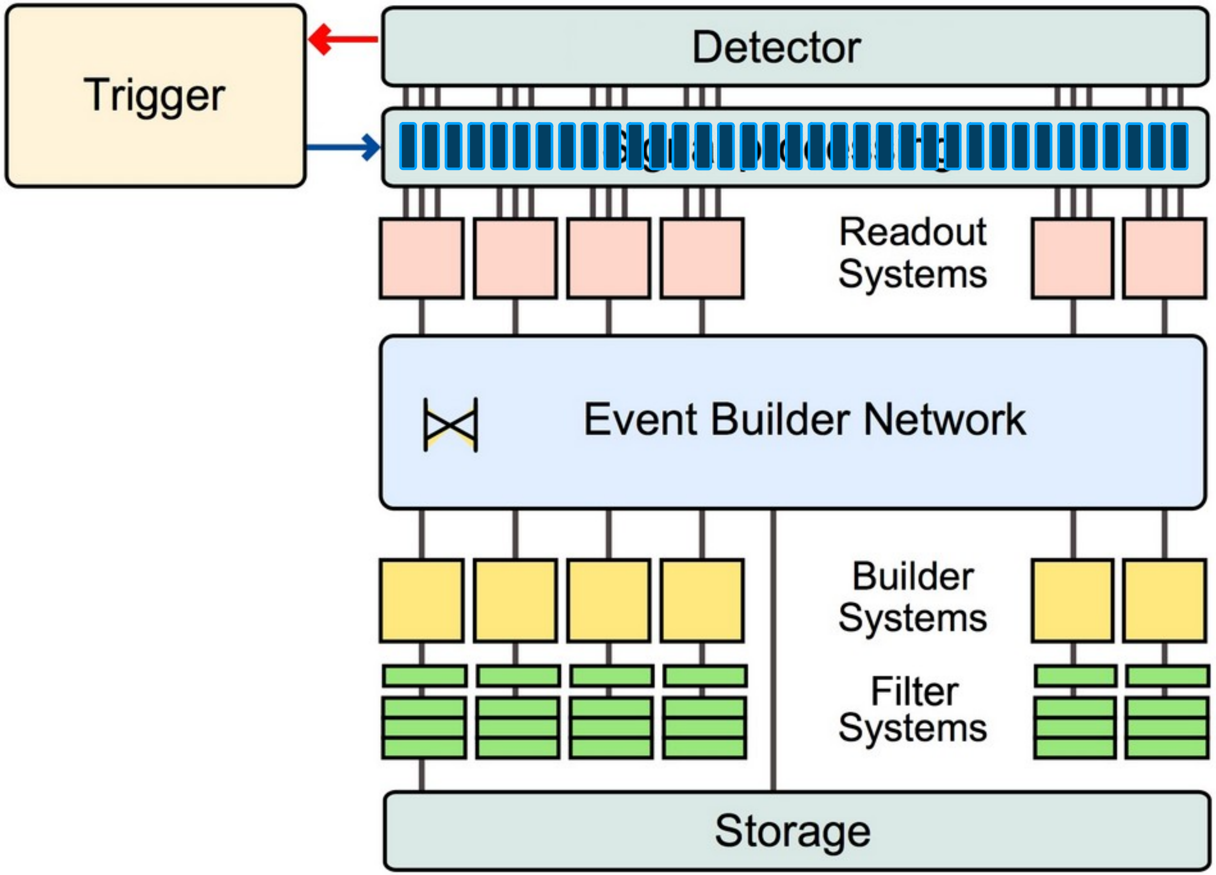
PUTTING IT ALL TOGETHER



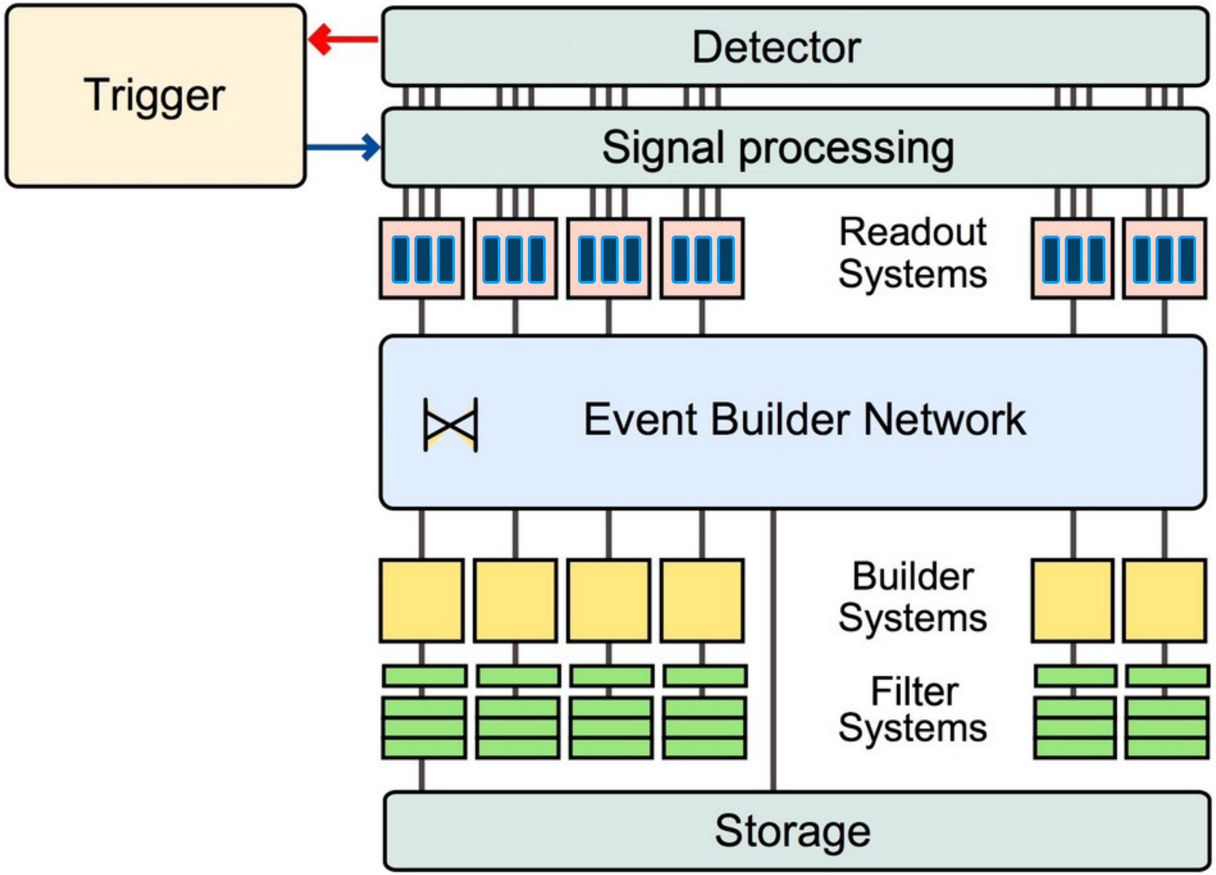
PUTTING IT ALL TOGETHER



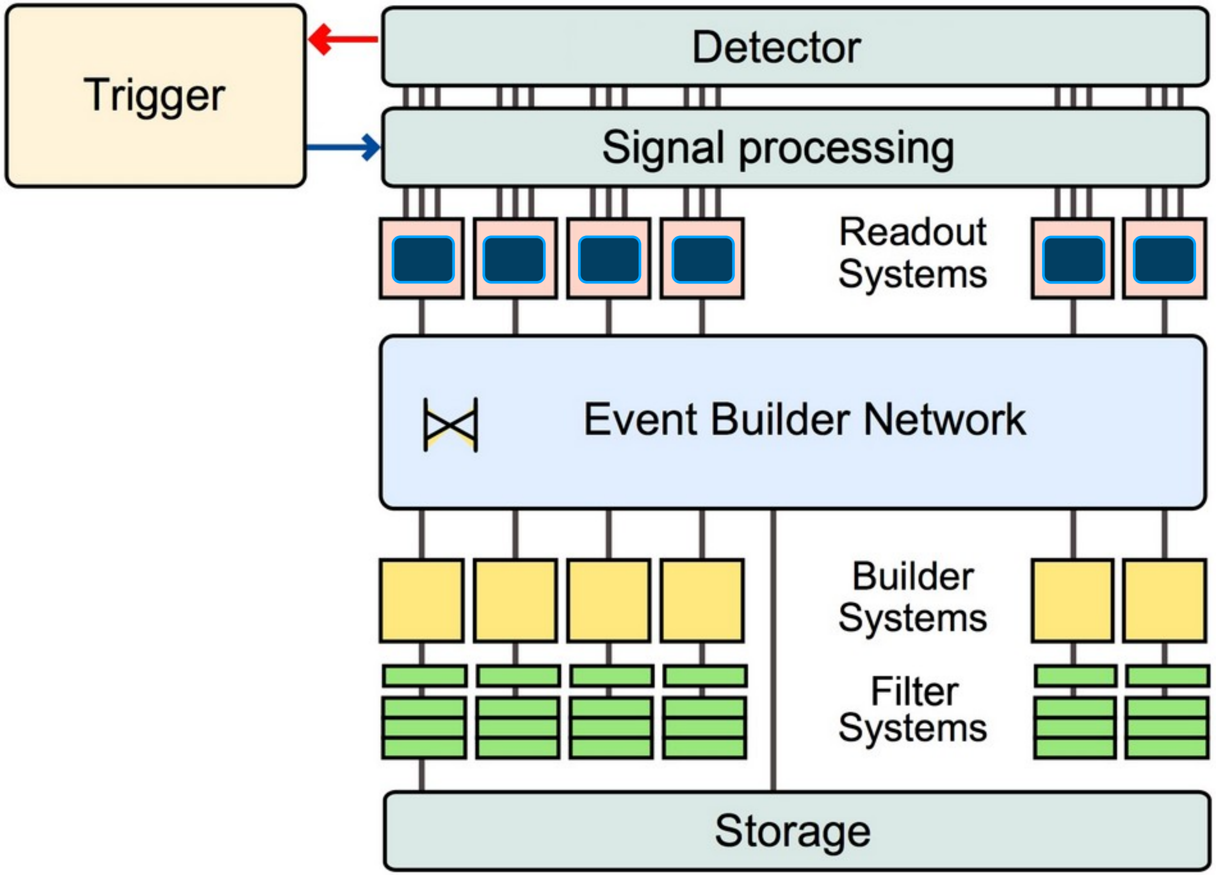
PUTTING IT ALL TOGETHER



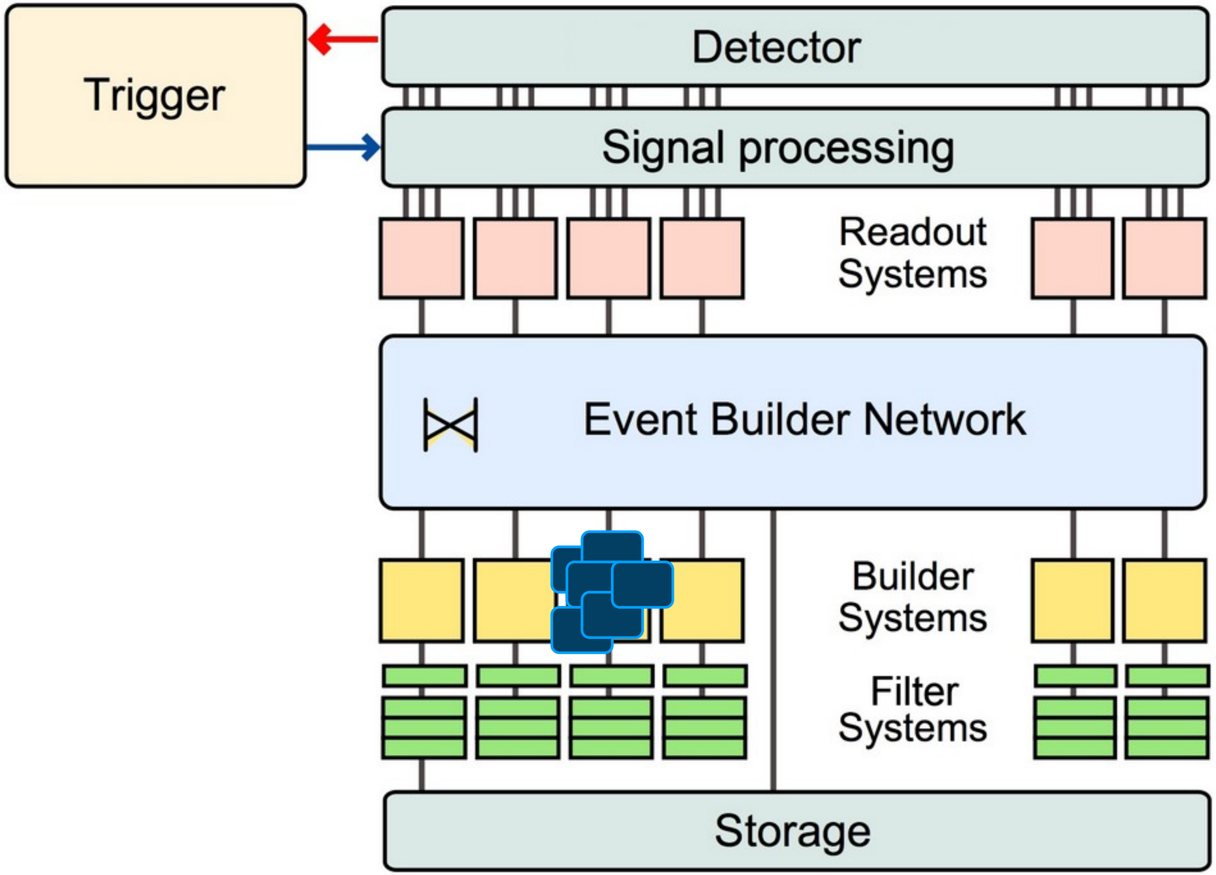
PUTTING IT ALL TOGETHER



PUTTING IT ALL TOGETHER

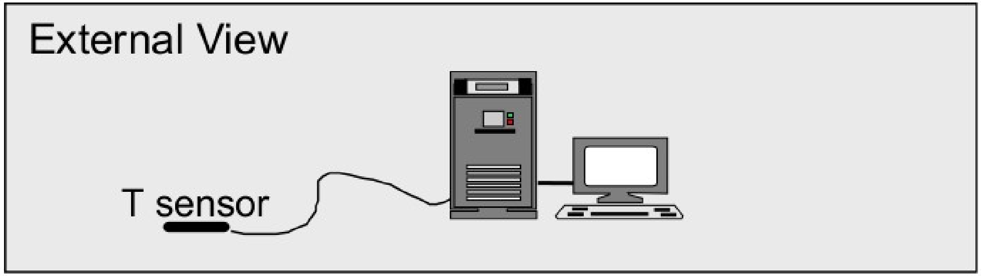


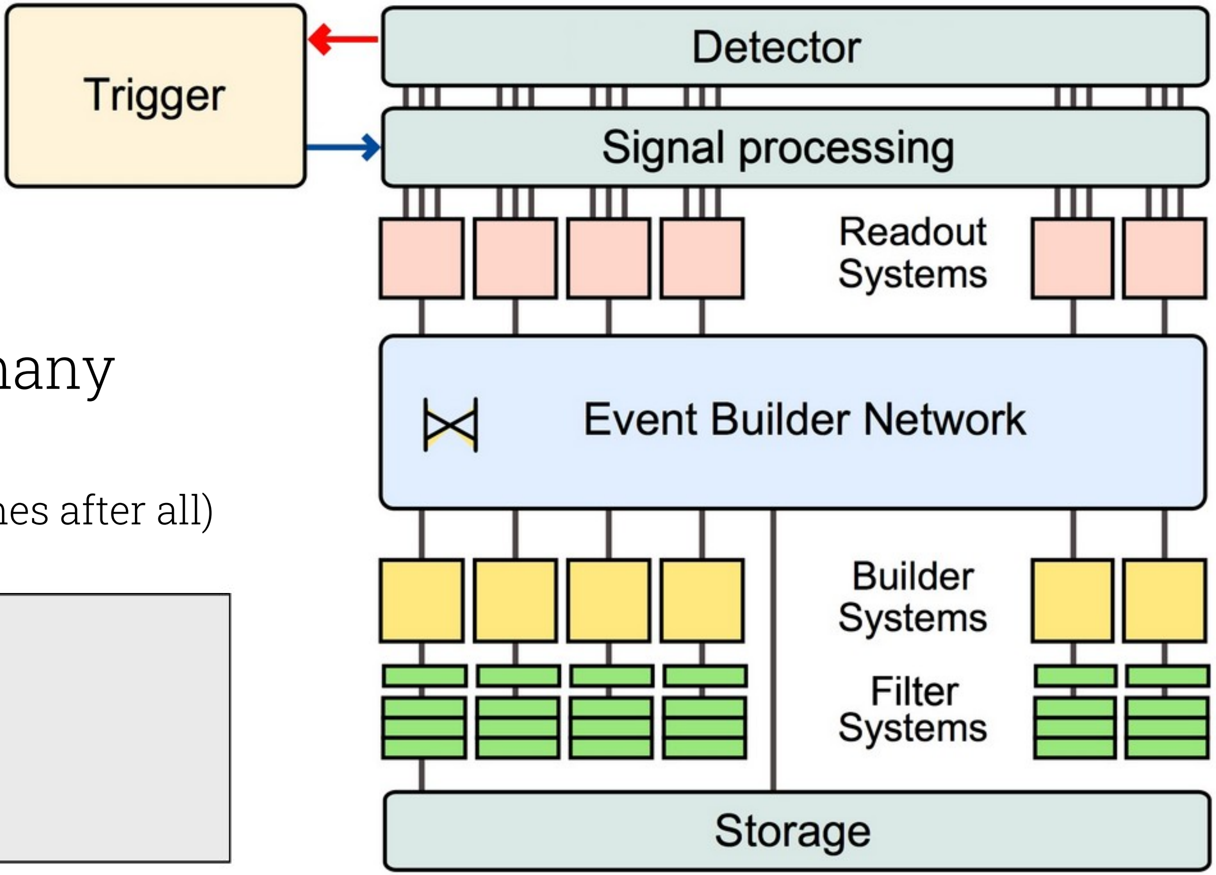
PUTTING IT ALL TOGETHER



SCALE

- In a large experiment, all of these systems are separate and have to be interconnected
- For smaller experiments, many functions can be combined
  (computers are general-purpose machines after all)

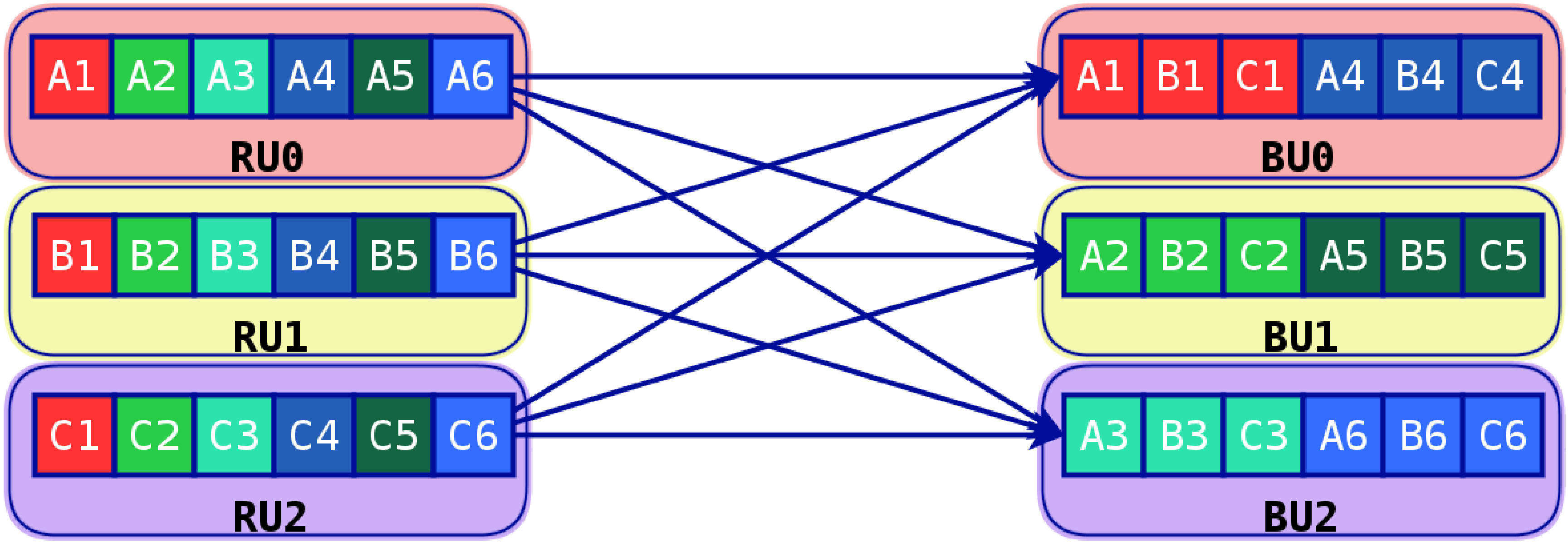




DATA COLLATION

OR: EVENT BUILDING

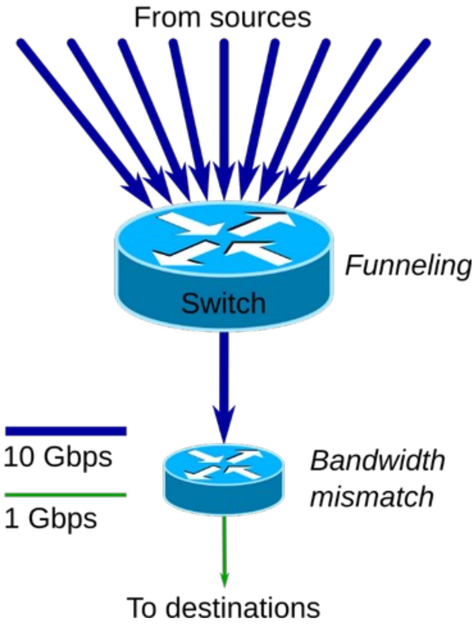
EVENT BUILDING IN THEORY

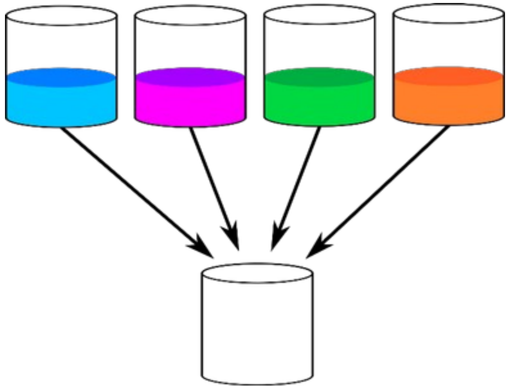


- Readout unit (RU): receives processed signals from some sensors
- Builder unit (BU): assembles all signals corresponding to the same observed phenomenon

EVENT BUILDING NETWORKS

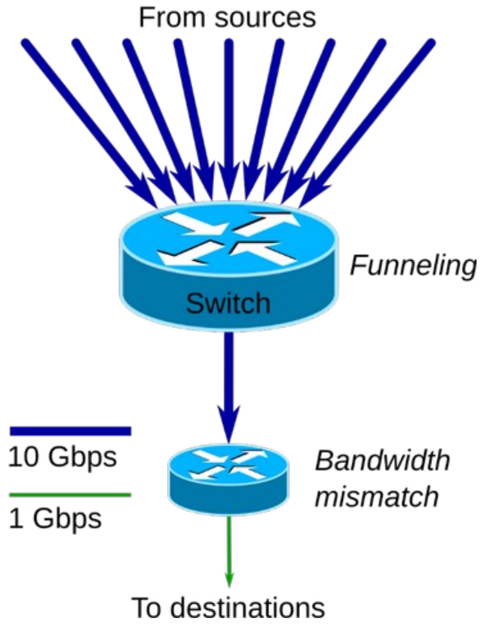
- The BUs collect data from different RUs
  → Many-to-one communication
- Data transfers are driven by the availability of the data from the detector
  → Synchronous, bursty traffic
- When many sources send synchronous microbursts of data to a destination
  → Congestion
  → The network buffers are overflown
- Must be kept under control, otherwise:
  "Catastrophic throughput collapse"

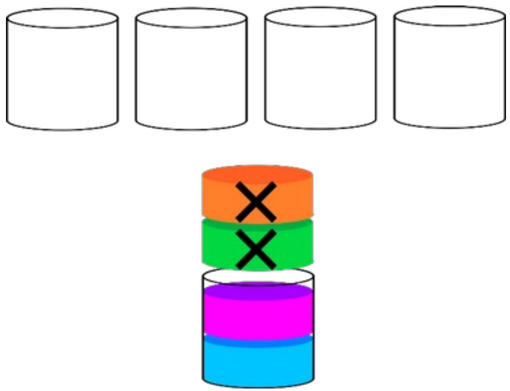




EVENT BUILDING NETWORKS

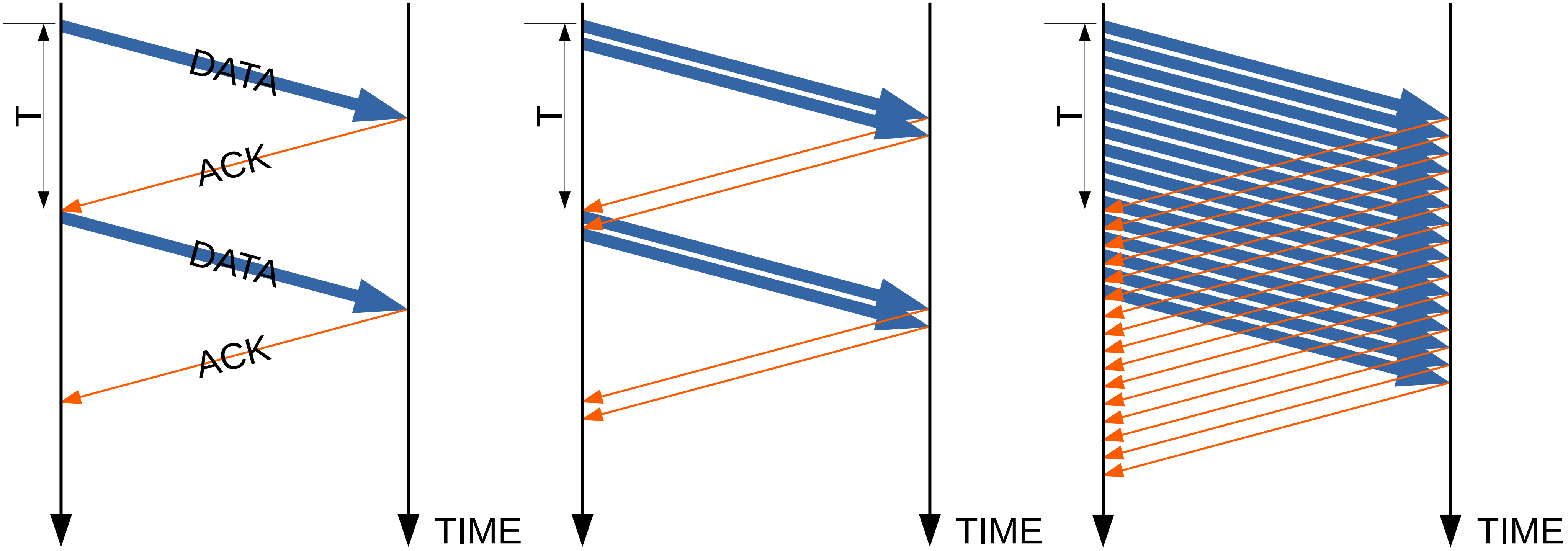
- The BUs collect data from different RUs
  → Many-to-one communication
- Data transfers are driven by the availability of the data from the detector
  → Synchronous, bursty traffic
- When many sources send synchronous microbursts of data to a destination
  → Congestion
  → The network buffers are overflown
- Must be kept under control, otherwise:
  "Catastrophic throughput collapse"





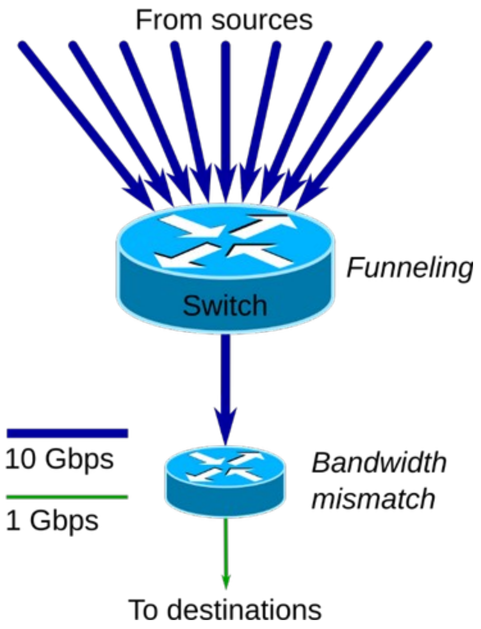
ACK-BASED CONGESTION CONTROL

B “in-flight” bytes | *2B* “in-flight” bytes | *enough* “in-flight” bytes

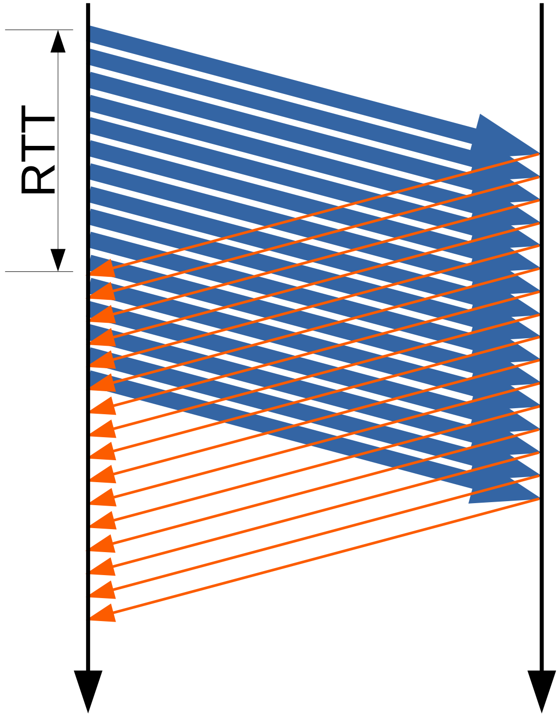


Throughput: *B*/*T* | Throughput: 2*B*/*T* | Throughput: 100%

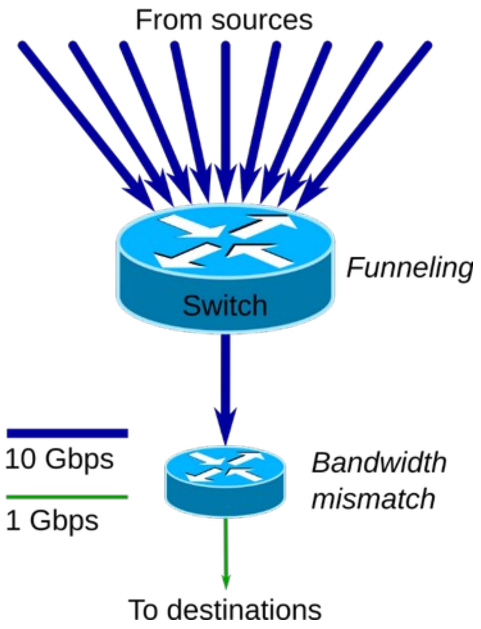
THROUGHPUT-RTT PRODUCT



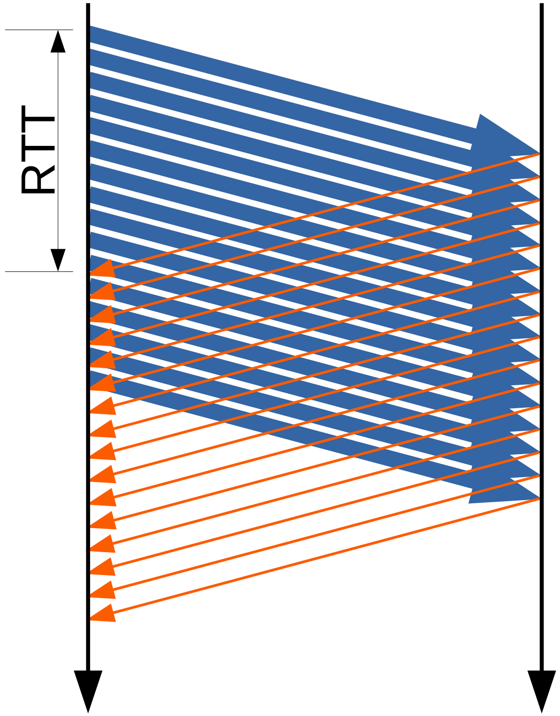
- What determines how many in-flight bytes are "enough"?
- The slowest / most used link!
- Calculation:
  - Minimum unused link throughput (B/s): R_{free}
  - Round-trip time: T_{RTT}
  - Optimal amount of in-flight packets: $R_{free} T_{RTT}$
  - A.k.a.: bandwidth-delay product (BDP)



LOCAL DECISIONS, GLOBAL IMPACT

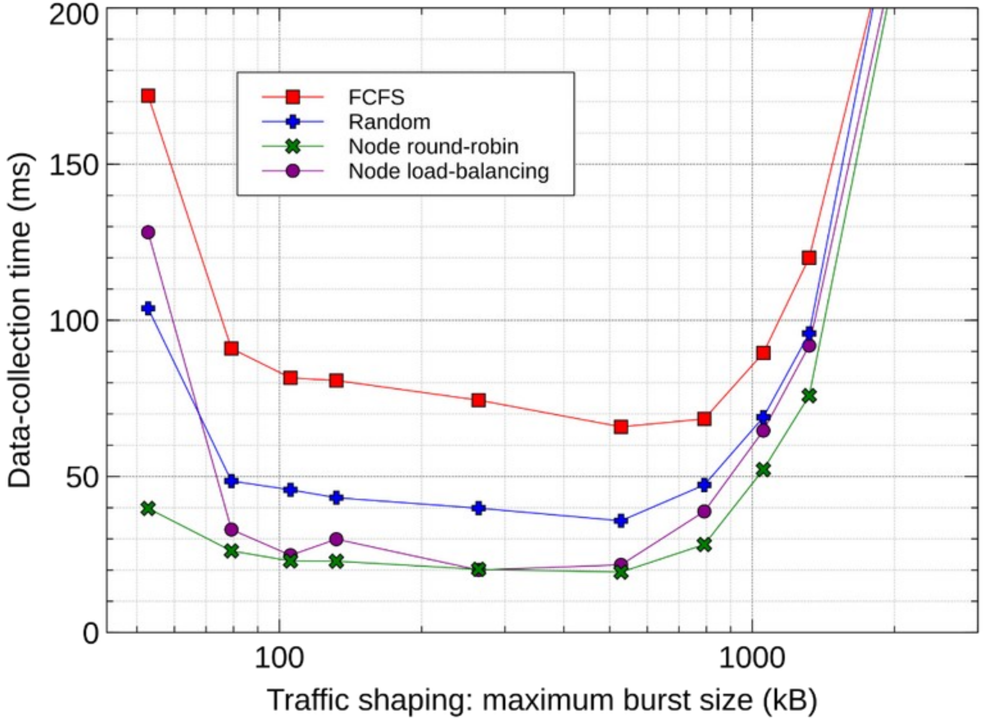


- Can a sender measure R_{free} ?
  Not really!
- Instead: gradually increase the amount of in-flight data until something goes wrong
- With many synchronous senders "something wrong" will occur at the same time for all of them
  → all of them will slow down (too much!)



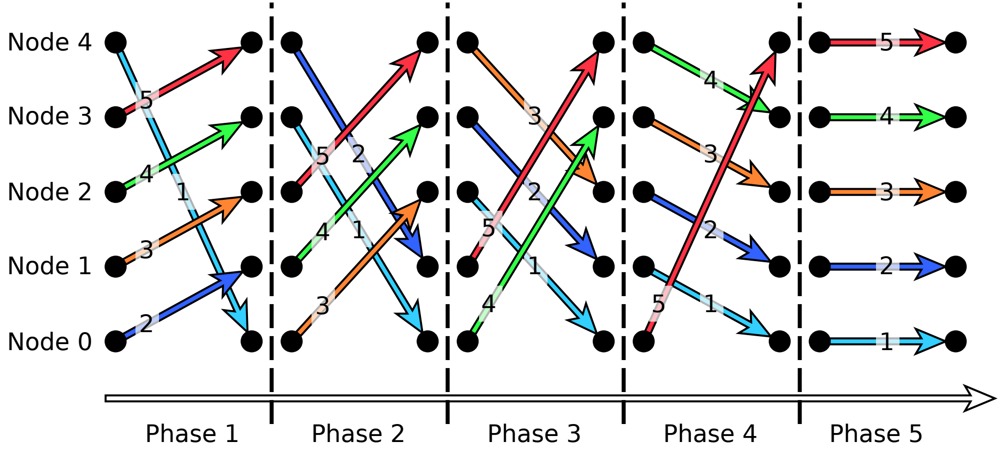
PULL-BASED TRANSFERS TO THE RESCUE

- In DAQ, we can precisely control how we use the network
- BU pulls data from RUs:
  Can prevent too many RUs from sending to the same BU at the same time
- Tuning needed:
  - Shaping too aggressive
    → bottleneck
  - Shaping too lax
    → congestion



EVEN STRICTER TRAFFIC SHAPING

- With N RUs, the building of N events is divided into N phases
- In every phase one RU sends data to one BU, and every BU receives data from one RU
- During phase n, RU m sends data to BU $(m+n) \mod N$
- All the units switch synchronously from phase n to phase $n+1$



- On the right network topology, this can avoid congestion altogether

BUFFERS, AGAIN

- Traffic shaping techniques require waiting for the “right” moment to send data into the network
- Waiting == buffering
- Thankfully, the RUs are computers outside of the detector
  - Very large buffers (RAM) are relatively cheap
  - No sensitive volume “stolen”

✓ Optimise trigger for low latency, data collection for high throughput

DATA FILTER

Left as an exercise for the ~~reader~~ user

STORAGE

ERASURE CODING

- Parity bit: count the 1s in a string of bits
  - Even number of 1s → Parity = 0
  - Odd number of 1s → Parity = 1
- Can be used to add redundancy without full copies

Bit 1	Bit 2	Bit 3	Parity
0	1	1	
0	0	0	
1	0	0	
1	1	1	

ERASURE CODING

- Parity bit: count the 1s in a string of bits
  - Even number of 1s → Parity = 0
  - Odd number of 1s → Parity = 1
- Can be used to add redundancy without full copies

Bit 1	Bit 2	Bit 3	Parity
0	1	1	0
0	0	0	0
1	0	0	1
1	1	1	1

ERASURE CODING

- Oh no! The bits in the third position were on a broken memory

Bit 1	Bit 2	~~Bit 3~~	Parity
0	1	~~1~~	0
0	0	~~0~~	0
1	0	~~0~~	1
1	1	~~1~~	1

ERASURE CODING

- Oh no! The bits in the third position were on a broken memory
- But we still have parity!

Parity of bit 1, bit 2, and the original parity

Bit 1	Bit 2	~~Bit 3~~	Parity	
0	1	~~1~~	0	1
0	0	~~0~~	0	0
1	0	~~0~~	1	0
1	1	~~1~~	1	1

ERASURE CODING

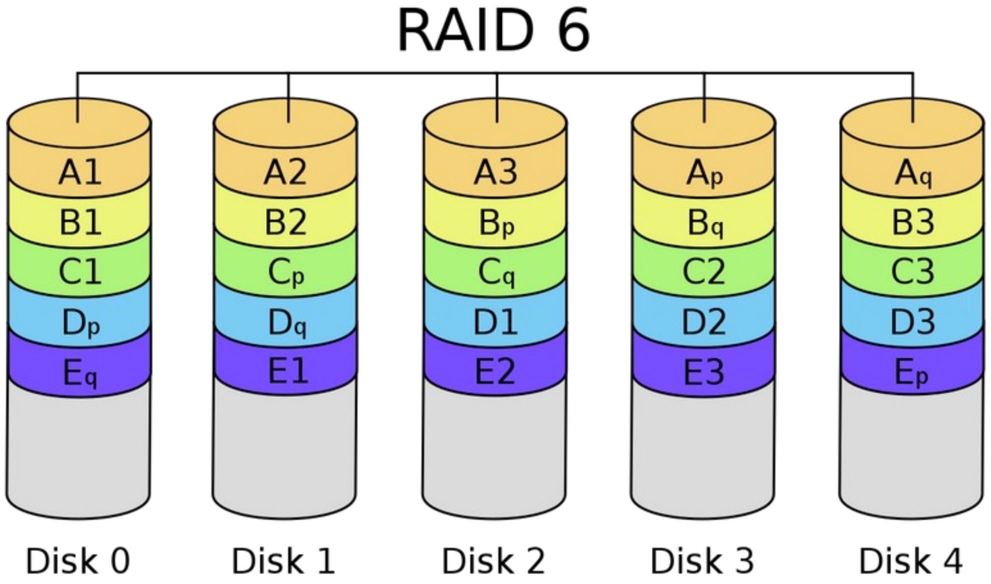
- Any of the original bits can be recovered this way
- If we lose more than one bit at the same time, we're out of luck, though

Bit 1	Bit 2	Bit 3	Parity
0	1	~~1~~	0
0	0	~~0~~	0
1	0	~~0~~	1
1	1	~~1~~	1

ERASURE CODING IS EVERYWHERE

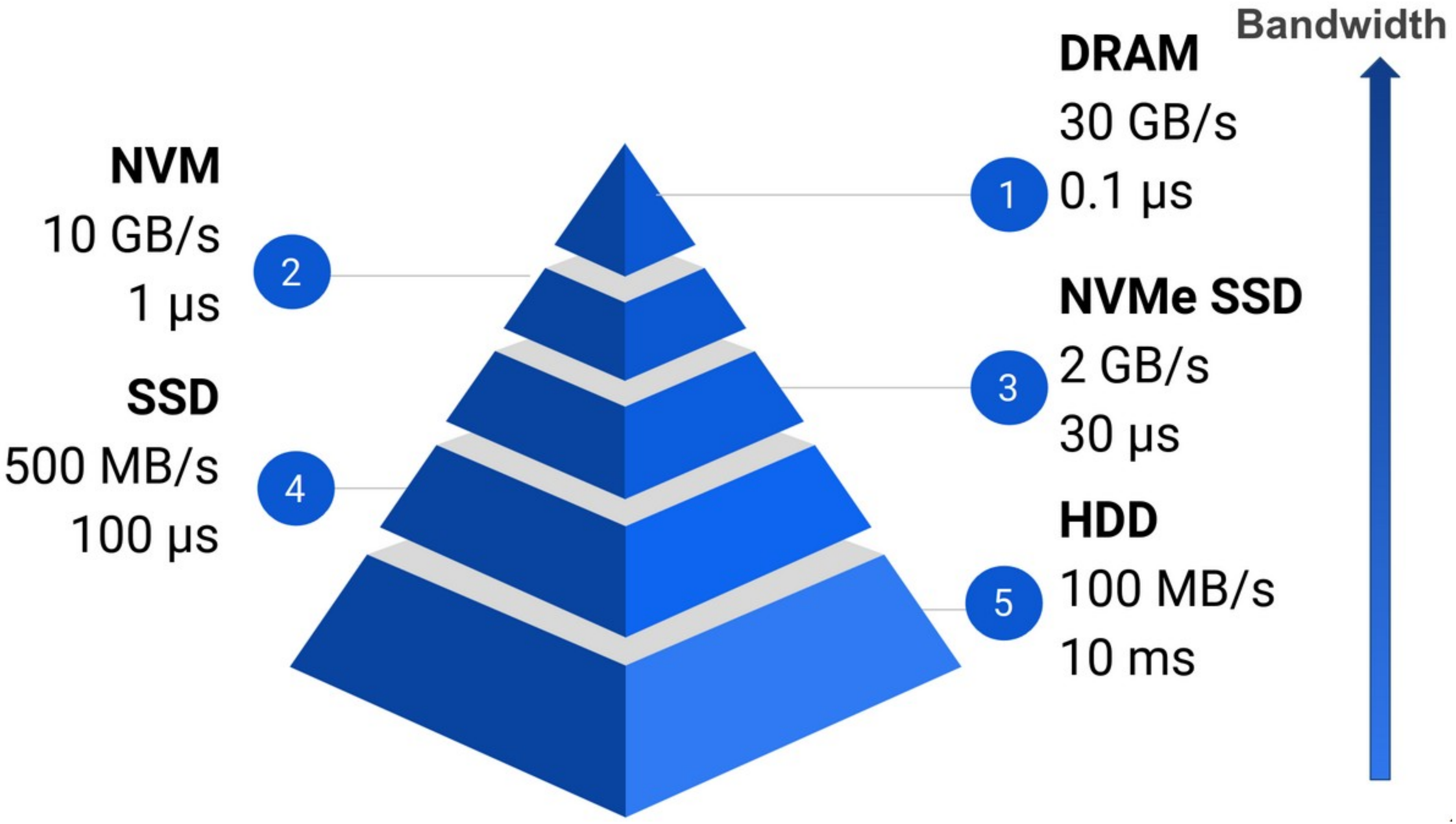
- Not limited to parity: whole families of error correcting codes exist
  - Operate on (and can recover) more than 1 bit
  - Can use more than one at the same time
  - Most common: Reed-Solomon

- Used in:
  - Many kinds of links (optical or not)
  - Storage (from RAM to hard disks)



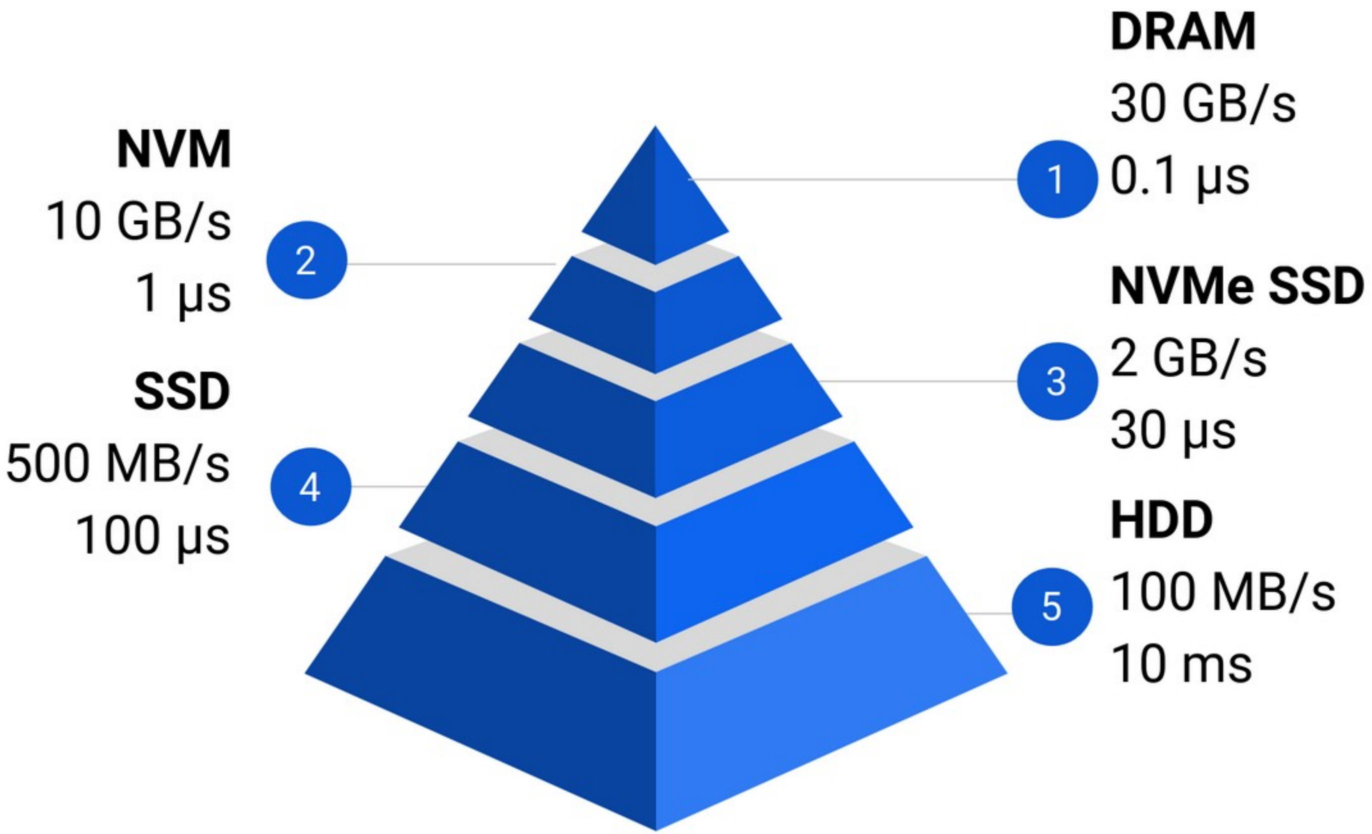
From C. Burnett, https://commons.wikimedia.org/wiki/File:RAID_6.svg

PICK THE RIGHT TECHNOLOGY



WILL I EVER SHUT UP ABOUT BUFFERS?

- Faster storage technologies can be used as derandomising buffers for slower but cheaper tech



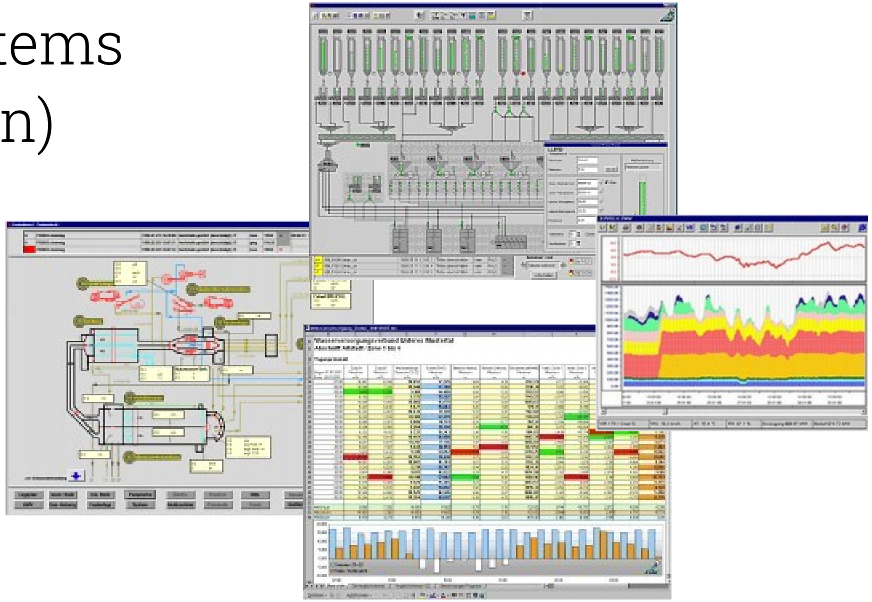
CONTROLS

ONE CONTROL SYSTEM TO RULE THEM ALL

- All parts of the experiment must work as one: a central "conductor" system is a must
- **Monitoring**: detect problems as soon as possible
- **Configuration**:
  - Get the experiment to the desired state
  - Sequencing and synchronisation of operations across components
- **Automation**:
  - Avoid human mistakes
  - Speed up standard procedures

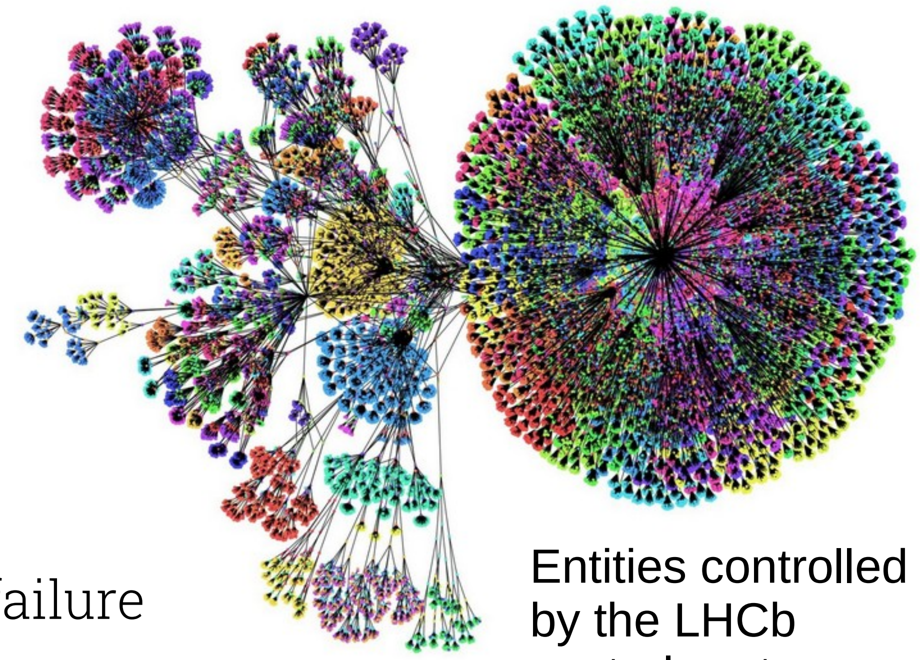
SCADA

- Can be based on commercial SCADA systems (Supervisory Control and Data Acquisition)
- Commonly used for:
  - Industrial automation
  - Control of factories, power plants, etc.
- Providing:
  - Run-time database
  - Display and archiving of monitoring data
  - Alarm definition and reporting tools
  - User interface design tools



SCALE, ONE LAST TIME

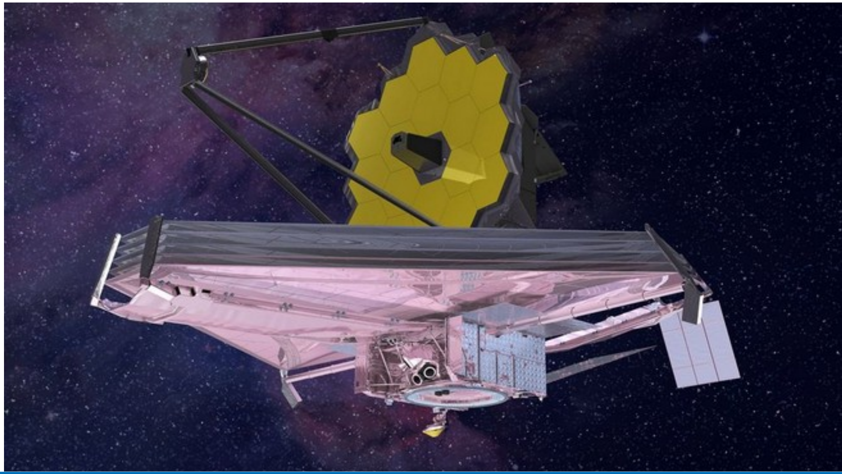
- In a large experiment, many independent low-probability faults can result in abysmal DAQ efficiency
- Example:
  - 1000 sensors
  - Each of them has a 0.1% probability of failure
  - Any failure stops the DAQ
  - Probability that the DAQ is stopped: 37%!



Entities controlled by the LHCb control system

SCALE, ONE LAST TIME

- In a large experiment, many independent low-probability faults can result in abysmal DAQ efficiency
- Failures should be non-fatal as much as possible
- Maintenance windows (i.e.: when the experiment is stopped to fix faults) heavily influence the design of the detector and DAQ



GRAZIE PER L’ATTENZIONE!

AND THANKS TO MY PREDECESSORS
N. NEUFELD, W. VANDELLI, R. FERRARI, E. MESCHI
FOR THE “INSPIRATION” I STOLE FROM THEIR LESSONS

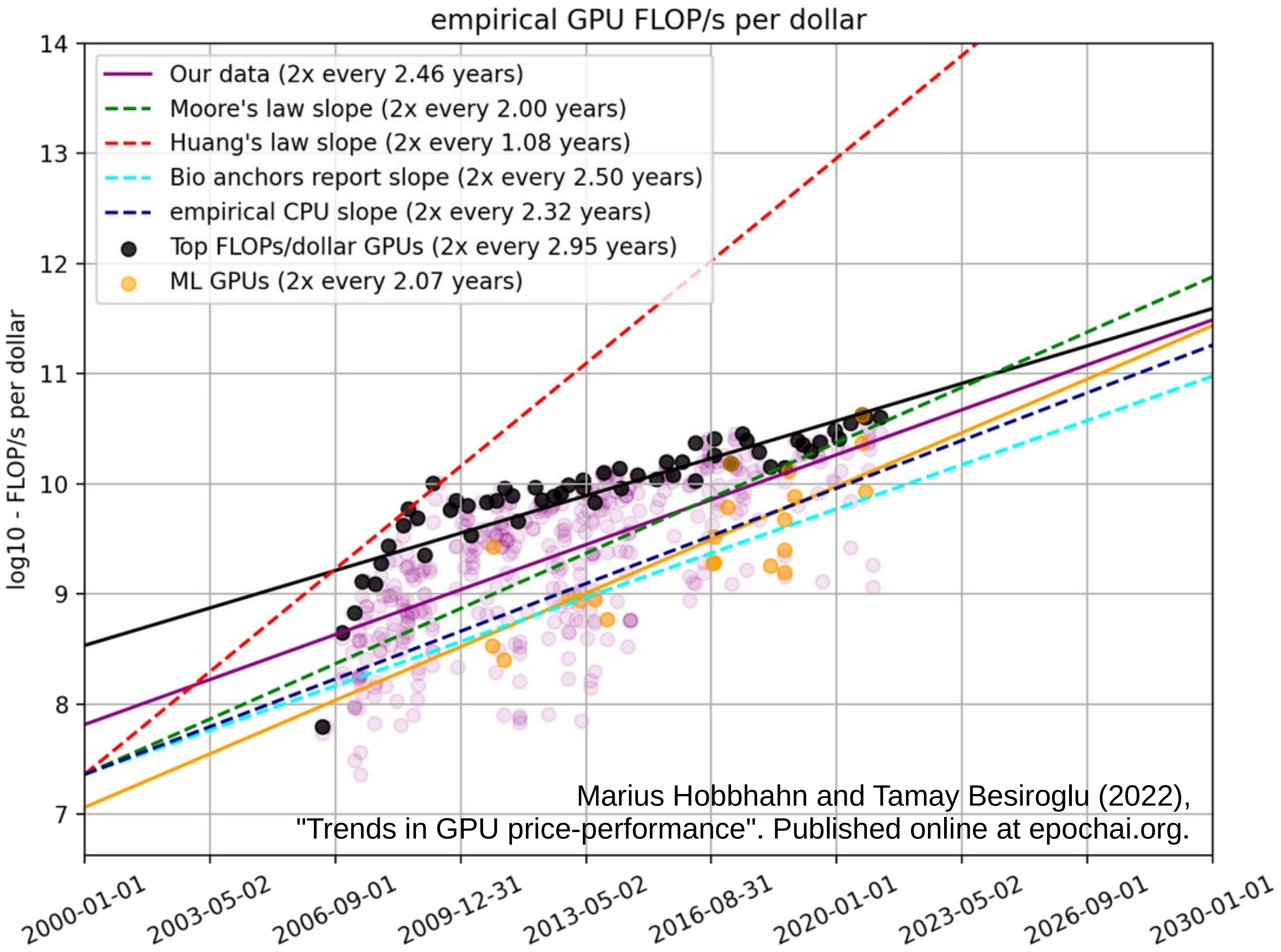
FOR MORE IN-DEPTH LESSONS AND LABS:
https://isotdaq-schools.web.cern.ch/

A GREAT INTRODUCTION TO DETECTOR ELECTRONICS:
https://www-physics.lbl.gov/~spieler/

BACKUP

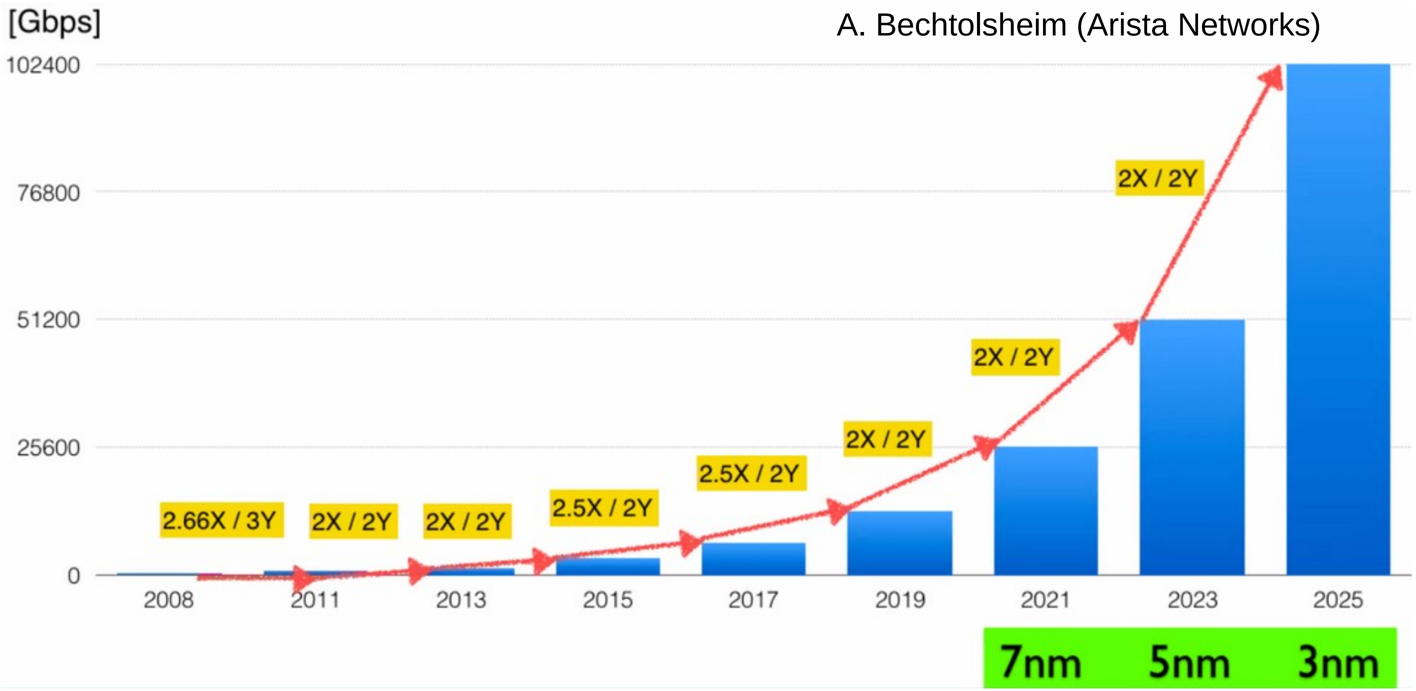
COMPUTE

Reports of the death of Moore's Law have been greatly exaggerated

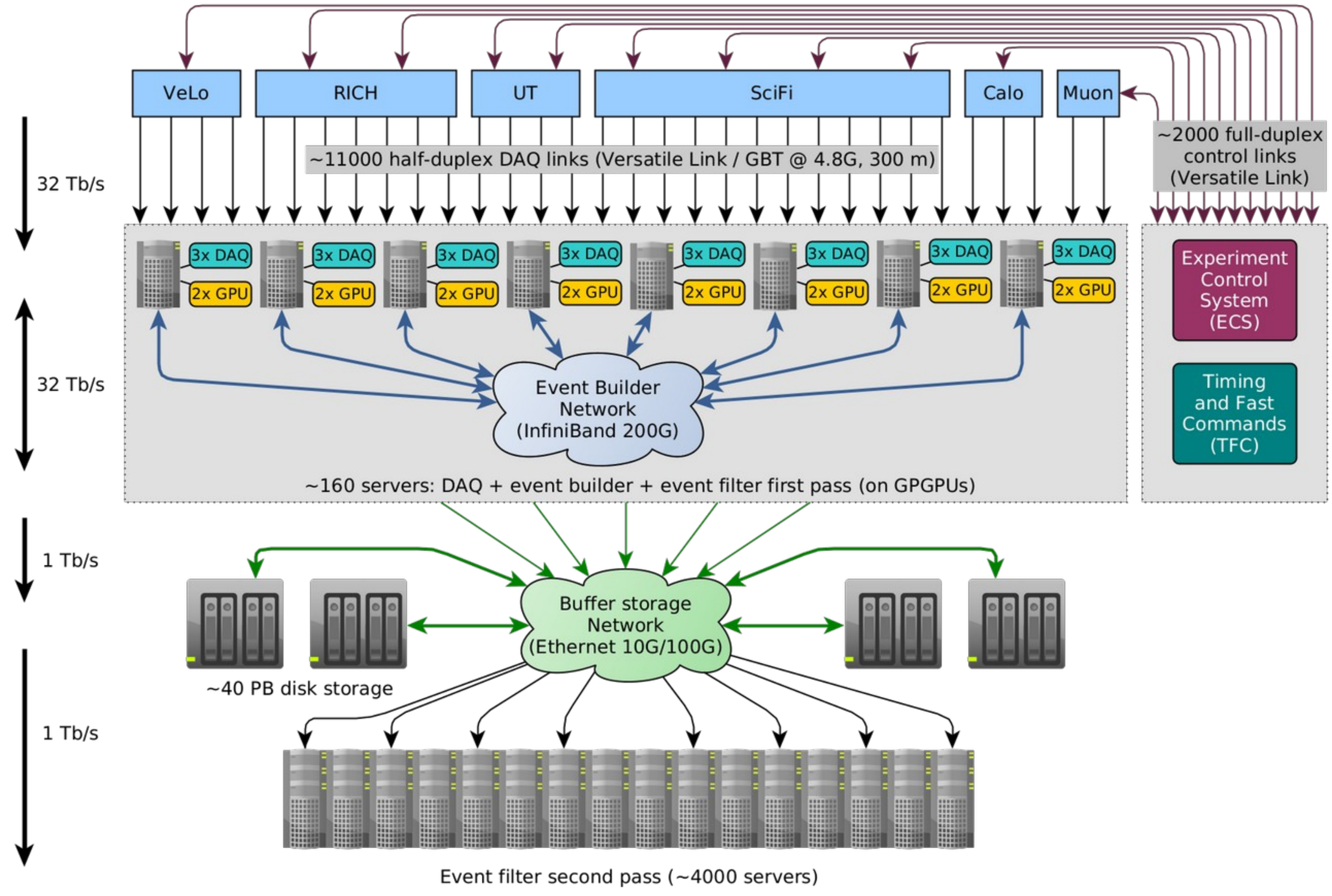


NETWORK

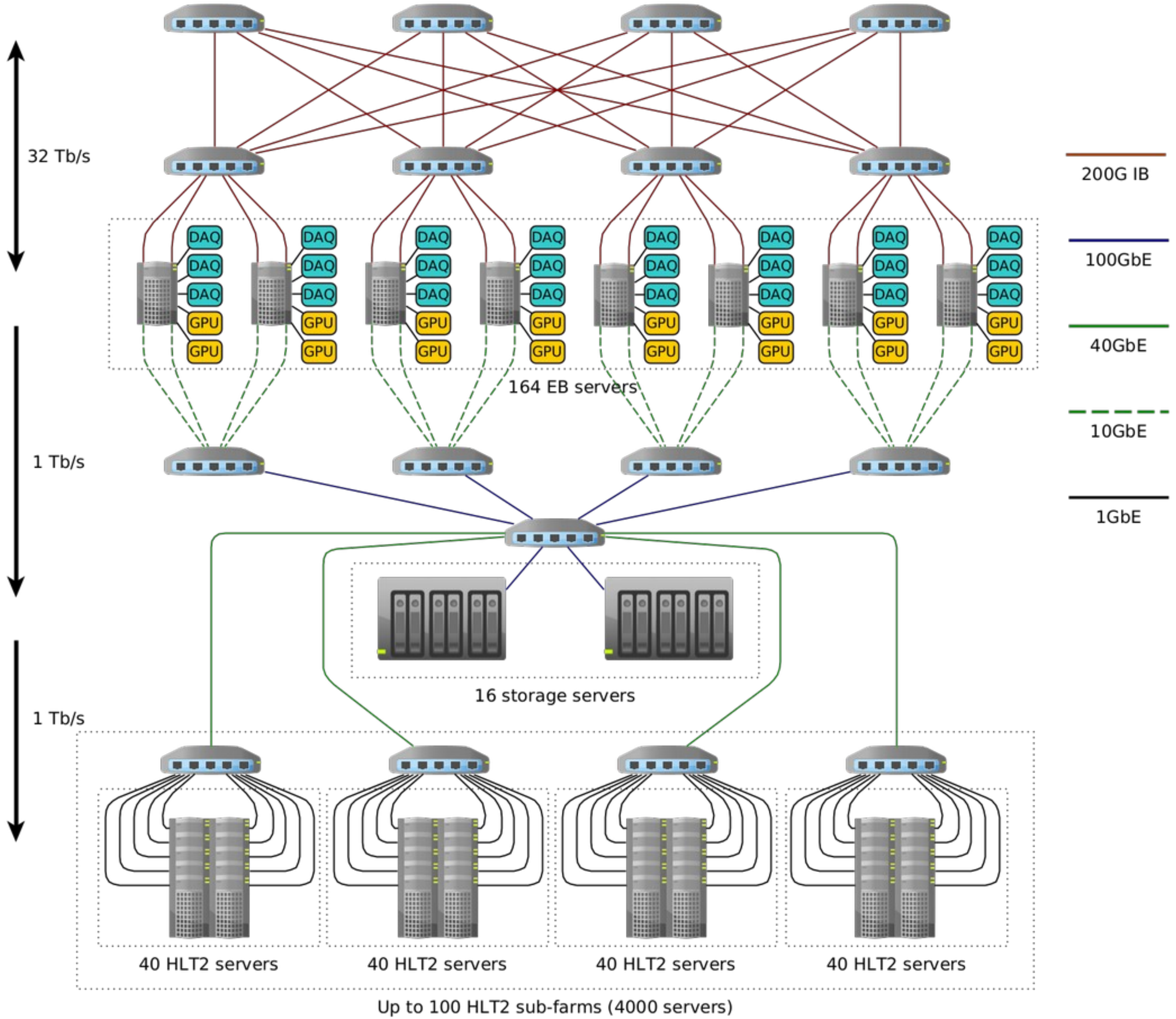
- 25 Tb/s **single-ASIC switches** available today
- 50 Tb/s is around the corner
- Evolution driven by cloud and ML



LHCb DAQ


VeLo
RICH
UT
SciFi
Calo
Muon
~11000 half-duplex DAQ links (Versatile Link / GBT @ 4.8G, 300 m)
~2000 full-duplex control links (Versatile Link)
32 Tb/s
3x DAQ
2x GPU
Event Builder Network (InfiniBand 200G)
32 Tb/s
~160 servers: DAQ + event builder + event filter first pass (on GPGPUs)
Experiment Control System (ECS)
Timing and Fast Commands (TFC)
1 Tb/s
Buffer storage Network (Ethernet 10G/100G)
~40 PB disk storage
1 Tb/s
Event filter second pass (~4000 servers)

LHCb DAQ


32 Tb/s
1 Tb/s
1 Tb/s
DAQ
GPU
164 EB servers
16 storage servers
40 HLT2 servers
40 HLT2 servers
40 HLT2 servers
40 HLT2 servers
Up to 100 HLT2 sub-farms (4000 servers)
200G IB
100GbE
40GbE
10GbE
1GbE