CMS Event-Builder Performance on State-of-the-Art Hardware

Remigius K Mommsen
Fermilab

on behalf of the CMS DAQ Group
Overview of DAQ for LHC Run 3 (2021-2024)

- **Storage**
- **Synchronous (40 MHz clock driven)**
- **Asynchronous (event driven)**

**Custom electronics**
- Detector Frontend
- Backend
- FEROL
- Event-builder network
- FEDbuilder/data to surface network
- RUBU
- HLT
- EVMBU
- HLT
- Event builder network
- Storage
Overview of DAQ for LHC Run 3 (2021-2024)

- Custom electronics
- Commercial equipment / COTS

- Synchronous (40 MHz clock driven)
- Asynchronous (event driven)

- ~750 backends
- 0.1 - 9 kB fragments
- @ 100 kHz L1 trigger rate
Overview of DAQ for LHC Run 3 (2021-2024)

~750 backends
0.1 - 9 kB fragments
@ 100 kHz L1 trigger rate

~460 FEROLs

Synchronous (40 MHz clock driven) Asynchronous (event driven)
Overview of DAQ for LHC Run 3 (2021-2024)

- Synchronous (40 MHz clock driven)
- Asynchronous (event driven)

~750 backends
0.1 - 9 kB fragments
@ 100 kHz L1 trigger rate

~460 FEROLs

~460 FEROLs

30-50 RUBU nodes
Overview of DAQ for LHC Run 3 (2021-2024)

- Storage
  - Synchronous (40 MHz clock driven)
  - Asynchronous (event driven)

- Custom electronics
- Commercial equipment / COTS

- Event-builder network
  - EVMBU
  - RUBU
  - HLT

- Detector
  - FEROL
  - ~460 FEROLs
  - ~750 backends
  - 0.1 - 9 kB fragments
  - @ 100 kHz L1 trigger rate

- 30-50 RUBU nodes

Synchronous (40 MHz clock driven) | Asynchronous (event driven)
Overview of DAQ for LHC Run 3 (2021-2024)

- ~750 backends
- 0.1 - 9 kB fragments @ 100 kHz L1 trigger rate

Custom electronics vs. Commercial equipment / COTS

- ~460 FEROLs
- ~460 backends

Event-builder network
- 30-50 RUBU nodes
- ~1100 HLT nodes
- ~32k physical cores
- Nvidia T4 GPUs
Overview of DAQ for LHC Run 3 (2021-2024)

- **Storage**
  - Synchronous (40 MHz clock driven)
  - Asynchronous (event driven)

- **Custom electronics**
  - EVMBU
  - FEDbuilder / data to surface network

- **Commercial equipment / COTS**
  - Event-builder network
  - FEROL
  - RUBU
  - HLT

- **Detector**
  - Frontend
  - Backend

- **Bits and Pieces**
  - ~750 backends
  - 0.1 - 9 kB fragments
  - @ 100 kHz L1 trigger rate

- **FIROL**
  - ~460 FIROLs

- **Event builder network**
  - ~30-50 RUBU nodes

- **HLT**
  - ~1100 HLT nodes
  - ~32k physical cores
  - Nvidia T4 GPUs

- **Storage**
  - ~20 GB/s and ~1.5 PB
Network Technologies
Network Technologies

**FEDbuilder**

- Data transferred from underground to surface over 200 m
- FEROLs use a simplified TCP/IP protocol to send data
- FEROL output streams aggregated from 10 Gbps into 100 Gbps Ethernet
Network Technologies

**FEDbuilder**
- Data transferred from underground to surface over 200 m
- FEROLs use a simplified TCP/IP protocol to send data
- FEROL output streams aggregated from 10 Gbps into 100 Gbps Ethernet

**Event-builder**
- Interconnects all RUBU nodes
- Used to build complete events
- 100 Gbps Infiniband (EDR) or Ethernet using RoCE (RDMA over Converged Ethernet)
Network Technologies

**FEDbuilder**
- Data transferred from underground to surface over 200 m
- FEROLs use a simplified TCP/IP protocol to send data
- FEROL output streams aggregated from 10 Gbps into 100 Gbps Ethernet

**Event-builder**
- Interconnects all RUBU nodes
- Used to build complete events
- 100 Gbps Infiniband (EDR) or Ethernet using RoCE (RDMA over Converged Ethernet)

**High-Level Trigger (HLT)**
- Interconnects RUBU node to a set of HLT nodes running the event selection
- Events read and written back using NFS
- 100 Gbps fanned out to 10 Gbps Ethernet network
Event-Builder Protocol

1. Event Request
2. Fragment
3. EVM
4. Super-fragment
5. Super-fragment
6. Event

FEROL
RU1
RU2
BU1
BU2

Assign event to BU1
Event-Builder Protocol

Event Manager — EVM

- Orchestrates the event building
- Receives the master fragment from TCDS
Event-Builder Protocol

Event Manager — EVM
- Orchestrates the event building
- Receives the master fragment from TCDS

Readout Unit — RU
- Receive TCP/IP streams from FEROLs
  - Number of streams depend on fragment sizes
- Checks data integrity (CRC) and sequence
- Buffers fragments and combines them into larger messages (super fragments)
Event-Builder Protocol

Event Manager — EVM
- Orchestrates the event building
- Receives the master fragment from TCDS

Readout Unit — RU
- Receive TCP/IP streams from FEROLs
  - Number of streams depend on fragment sizes
- Checks data integrity (CRC) and sequence
- Buffers fragments and combines them into larger messages (super fragments)

Builder Unit — BU
- Builds complete events
- Checks consistency of event
- Writes events to files on local RAM disk
DAQ3 Testbed (daq3val)

Readout/Builder Unit (RUBU)
- Dell R740 dual skylake & 288 GB RAM
  - 14 RUBUs with 16-cores @ 2.1 GHz
  - 2 RUBUs with 14-cores @ 2.6 GHz
- 1 Dell R6515 with single AMD 7502P (EPYC ROME) @ 2.5 GHz & 251 GB RAM
- ConnectX-5 NICs (dual and single port)

Ethernet networks
- Juniper QFX10002-72Q
  - 288 x 10 Gbps or 24 x 100 Gbps
  - 16 GB buffer
- Juniper QFX5200-32C
  - 32 x 100 Gbps
  - 16 MB buffer

Infiniband EDR
- Mellanox SB7800
  - 36 x 100 Gbps
All to All

Evolve 1 fragment with 1kB (TCDS)

Emulate 27 fragments on 14 RUs

Evolve builder network

 CMS Preliminary

<table>
<thead>
<tr>
<th>Fragment Size (bytes)</th>
<th>Throughput on RU (GB/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>300</td>
<td>1</td>
</tr>
<tr>
<td>1000</td>
<td>2</td>
</tr>
<tr>
<td>2000</td>
<td>3</td>
</tr>
<tr>
<td>3000</td>
<td>4</td>
</tr>
<tr>
<td>10000</td>
<td>5</td>
</tr>
</tbody>
</table>

- Infiniband EDR
- RoCE over 100 Gbps
Scaling Number of Input Streams

![Diagram showing the scaling of input streams from FEROL to EVMBU and RUBU, with throughputs measured in GB/s and event rates at EVM.](image)

- **FEROL** to **EVMBU**: 1 stream @ 1kB (TCDS)
- **FEROL** to **RUBU**: 8 - 44 streams

Graph showing throughput on RU (GB/s) and event rate at EVM (kHz) for different numbers of streams (8, 10, 16, 20, 27, 36, 44) with fragment sizes ranging from 300 to 10000 bytes.
Performance of One RUBU Node

A HLT node runs 6 HLT processes with 4 threads each
Throughput of Disk Writing

Events are written to files on a RAM disk

- Several threads build and write events concurrently into separate files
- Each file contains 100 events

Throughput is limited by size of memory pages

- Move to kernel 4 which supports huge memory pages for RAM disk

Further improvement by splitting event building and disk writing task into separate threads

- 6 builder threads
- 7 writer threads
Cache Misses

Observed large performance drop when moving from kernel 3.10 to kernel 4.18

- TCP buffer handling in kernel has changed
- Event fragments no longer in cache when re-calculating CRC in input stage

Improved performance

- Tuning of NUMA settings to assure locality
- Reduced FIFO length between input and fragment handling code
2nd generation EPYC architecture (“Rome”)

- Multiple dies interconnected by AMD Infinity Fabric
- I/O die with 128 PCIe 4.0 lanes
- 8 memory channels @ 3.2 GHz

Single socket CPU provides up to 64 cores

- No need to use dual CPUs as for Intel
- Less worry about NUMA as all I/O on single socket
- Still need to deal with NUMA inside chip

AMD vs Intel Skylake

- **Event Builder Network**
- **Event Builder**
- **EVMBU**
- **RUBU**
- **FEDbuilder**
- **FEROL**

**Throughput on RU (GB/s)**

**CMS Preliminary**

- **EVMBU + 1 RUBU**
  - 27 streams per RU

- **Event Size (kB)**
  - 0
  - 2
  - 4
  - 6
  - 8
  - 10
  - 12

- **Throughput on BU (GB/s)**
  - **EVM + 1 RUBU + 13 RUs**
  - **CMS Preliminary**
  - **1 + 1*27 + 13*27 fragments**

- **Event Rate at EVM (kHz)**
  - 0
  - 100
  - 200
  - 300
  - 400
  - 500

**Skylake**
- Skylake (disabled NUMA settings)

**AMD**
- AMD (no NUMA tuning)
Summary

Data acquisitions system for LHC run 3 will be based on recent hardware

- 100 Gbps Ethernet Networks with TCP/IP and RoCE
- RoCE offers similar performance at lower cost than Infiniband

High-end servers with 30-40 cores can handle TCP/IP, event-builder traffic and NFS

- Allows to reduce the DAQ system by a factor ~4 compared to the run 2 system
- Throughput limited by software overhead in handling fragment rate from TCP streams
- AMD single-socket EPYC ROME is an interesting alternative, but requires more studies

Kernel 4.18 required to meet performance goals

- Huge memory pages for RAMdisk are a game changer
- Some interesting surprises encountered along the way