Benchmarking Data Acquisition event building network performance for the ATLAS HL-LHC upgrade

Eukeni Pozo Astigarraga (CERN), Matias Bonaventura (CERN), Rodrigo Castro (UBA), Giacomo Levrini (INFN), James Maple (CERN), Ezequiel Pecker (UBA)

NETADMINS
ATLAS TDAQ

ATLAS TDAQ
Dataflow

SED SIMULACIÓN de EVENTOS DISCRETOS

Outline

- High-Luminosity LHC (HL-LHC) ATLAS Data Acquisition System
  - The Event Filter network switch requirements
- Building a Data Acquisition system prototype
- Achievable event rate
- Benchmarking a shallow shared buffer switch
  - Buffer scan I: lab
  - Buffer scan II: analytic model
  - Buffer scan III: simulation model
- Conclusions and outlook

HL-LHC ATLAS Data Acquisition system

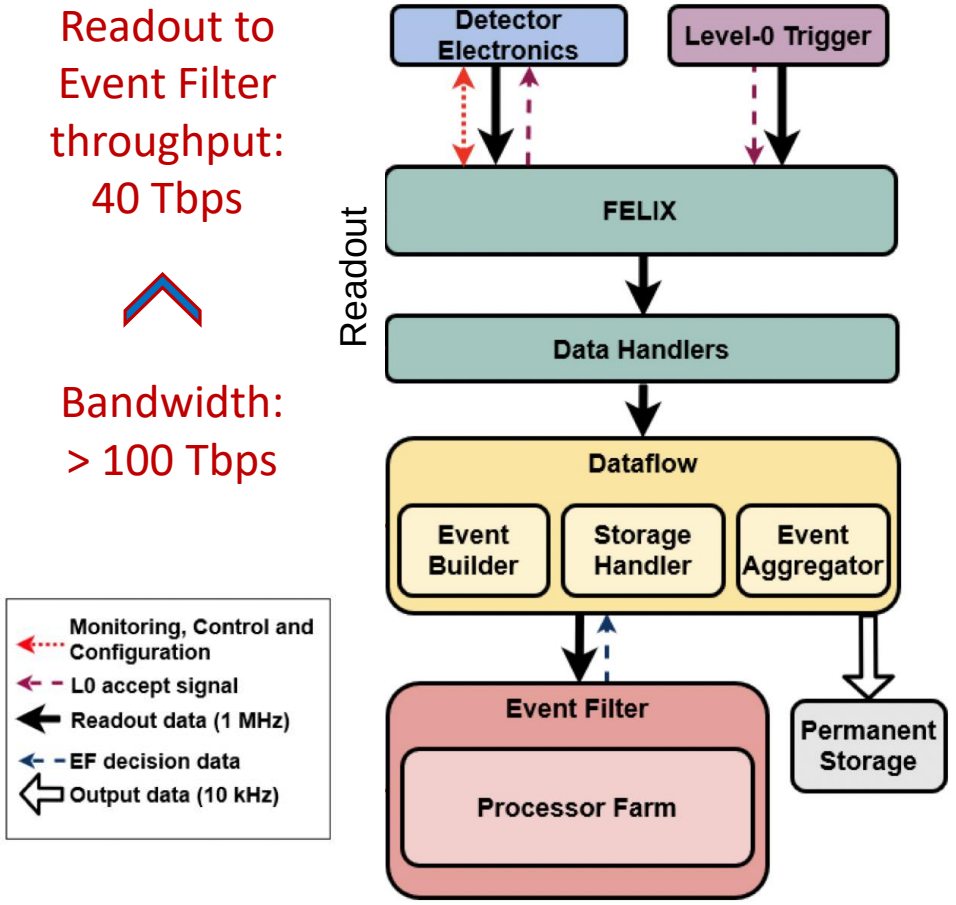
- New system requirements:
  - **Full Event Building** (EB) at 1 MHz L0 Trigger rate
  - Estimated **Event Size** for pile-up 140: 5 MB

Readout to Event Filter throughput: 40 Tbps

- Baseline scenario*: *CPU-based* Event Filter (EF) farm
  - Thousands of servers hosted in up to 100 racks
  - Each server equipped with a 25 Gbps link

Bandwidth: > 100 Tbps

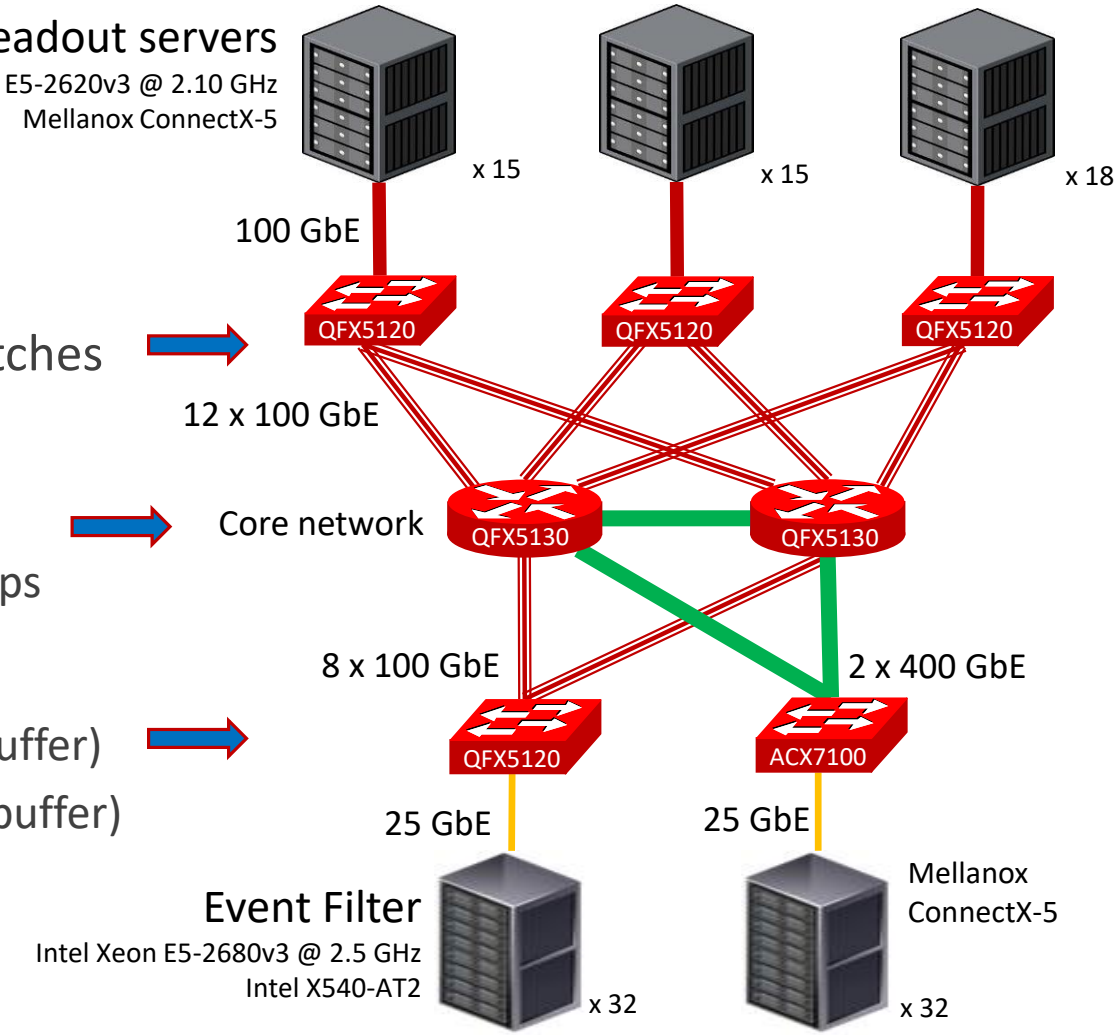
- **Requirements** for the top-of-rack (ToR) switches:
  - **Uplink bandwidth** to sustain a throughput of 500 Gbps
  - **Buffer** pool to absorb 12 kHz event-data rate



(*) Alternative scenarios based on accelerators are under study
They require less servers and consequently higher connectivity per server, reducing the stress on the network device buffers

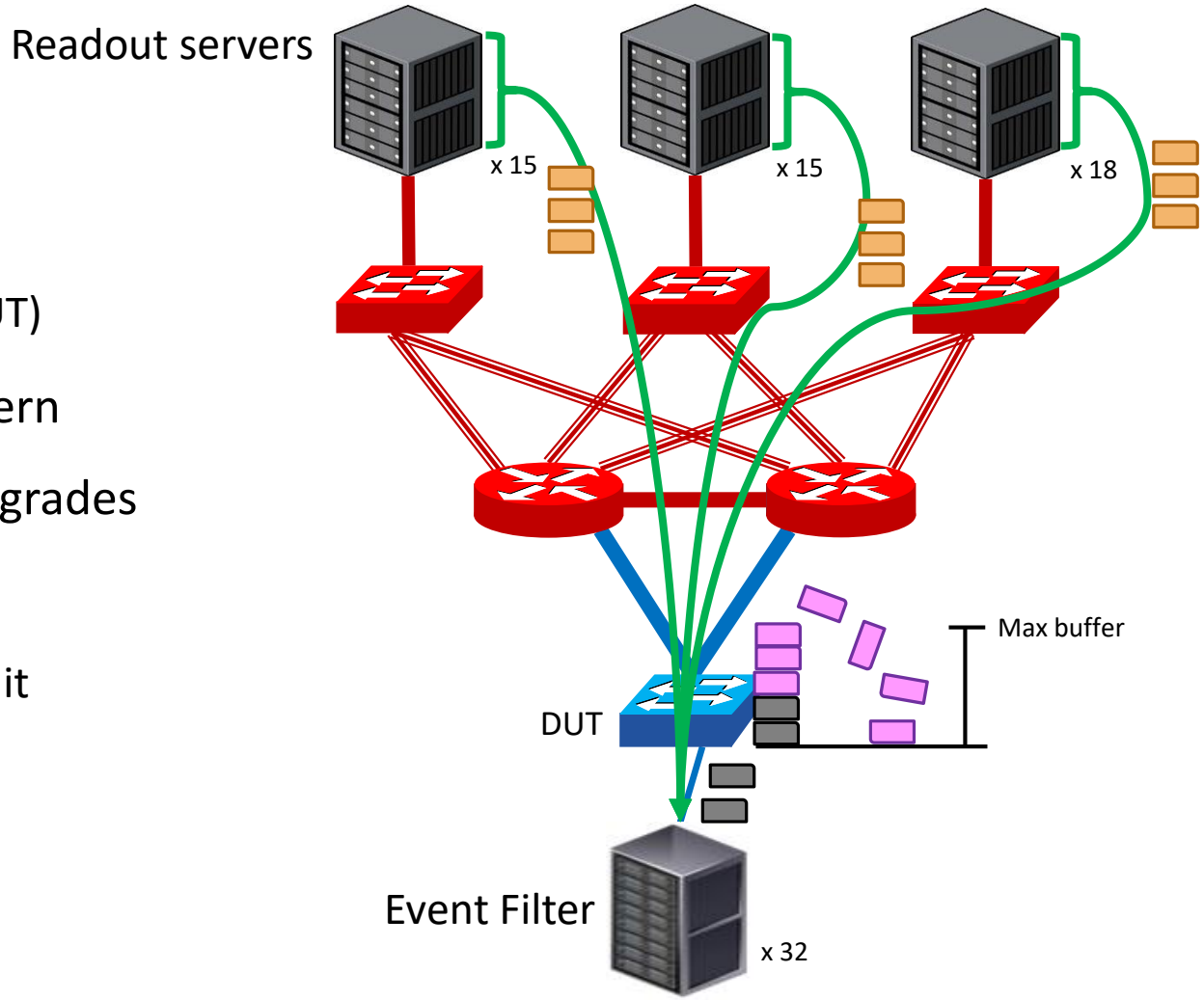
Data Acquisition System prototype

- Scaled down prototype
  - Final link speeds and topology
- Tuned LHC Run3 dataflow software
- No bottleneck on the Readout aggregation switches
  - Ingress bandwidth ~= Egress bandwidth
- 50 Tbps capable core network
  - Shallow buffers configured to suppress packet drops
- 2 different ToR switches tested:
  - **Juniper QFX5120:** BCM Trident 3 (27 MB shared buffer)
  - Juniper ACX7100: BCM Jericho 2 (8 GB dedicated buffer)



System dataflow and *TCP incast*

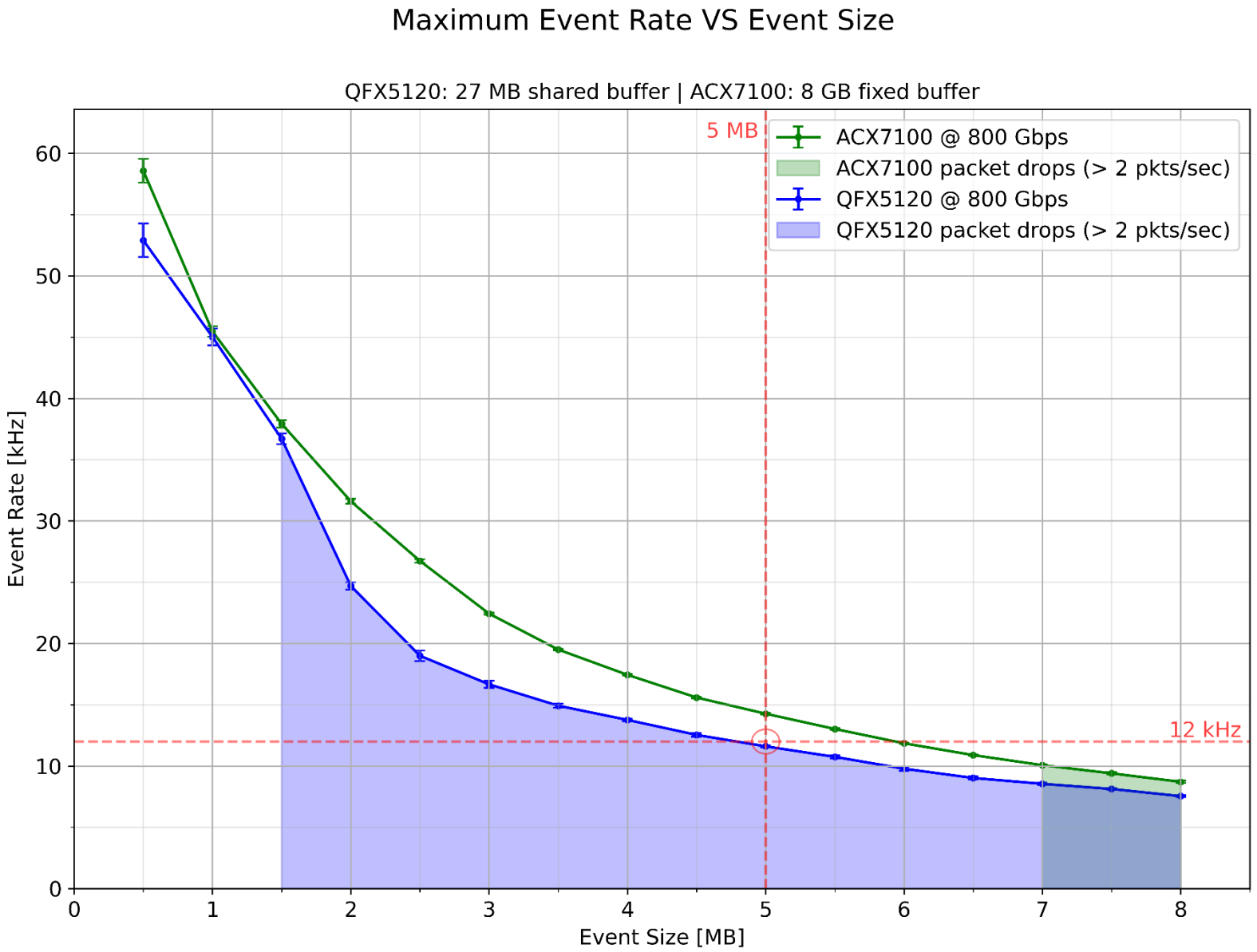
- 48 Readout servers acting as data sources
  - Sending event fragments upon request
- 32 Event Filter nodes as data sinks
  - Directly connected to the *Device Under Test* (DUT)
- Highly synchronized many-to-one traffic pattern
- A network pathology known as ***TCP incast*** degrades system performance
  - Instantaneous buffer oversubscription
  - Nothing TCP congestion control can do to avoid it



Event Rate & TCP incast

Event size scan for the 2 ToR switches:

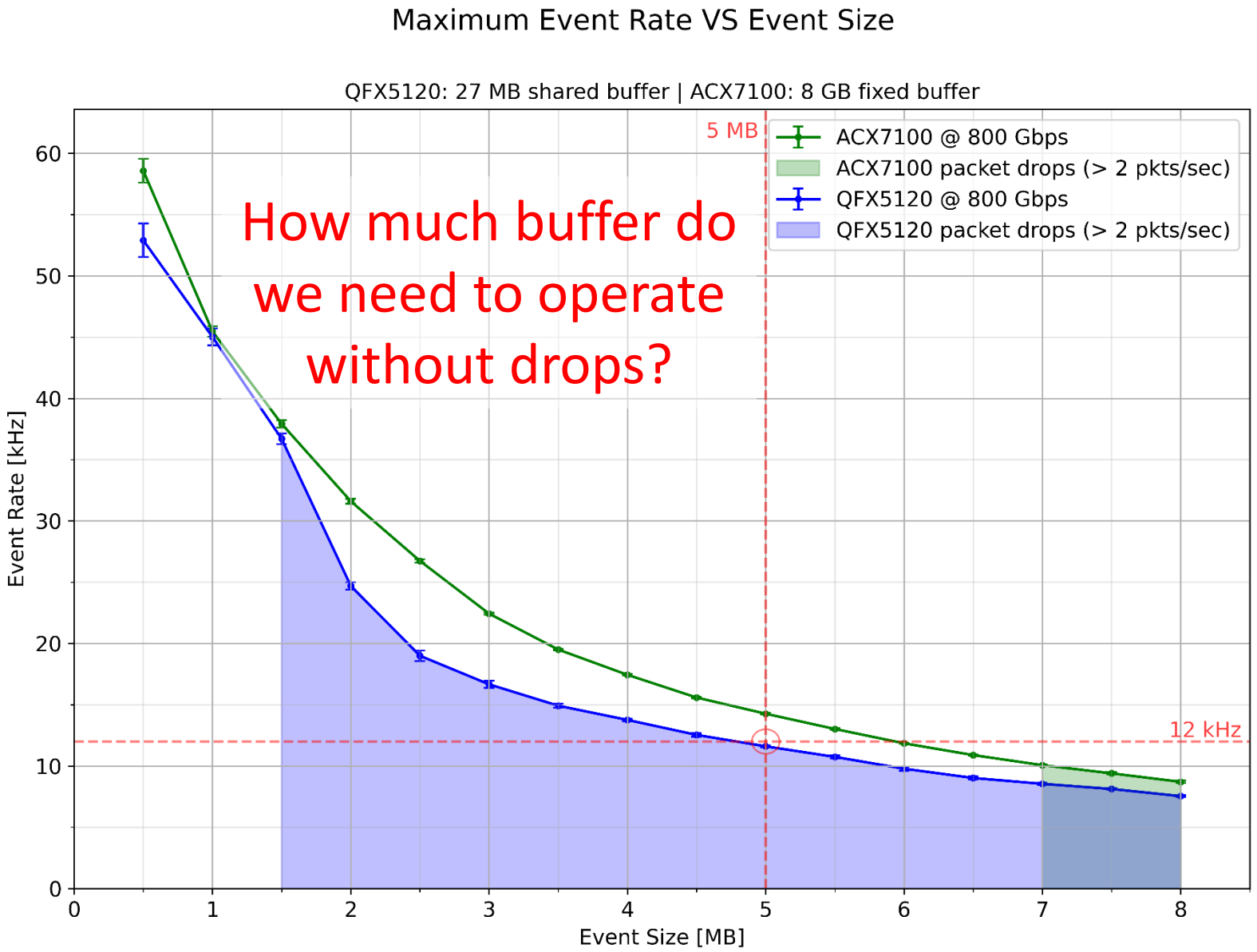
- ACX7100: achieves target event rate
  - Deep buffer device
  - 2.5 times the cost
- QFX5120: can't reach the required rate **without packet drops**
  - *TCP incast* prevents full utilization of link capacity



Event Rate & TCP incast

Event size scan for the 2 ToR switches:

- ACX7100: achieves target event rate
  - Deep buffer device
  - 2.5 times the cost
- QFX5120: can't reach the required rate **without packet drops**
  - *TCP incast* prevents full utilization of link capacity



Benchmarking a shallow shared buffer ToR switch

Candidate platform: Juniper QFX5120 (Broadcom Trident 3)

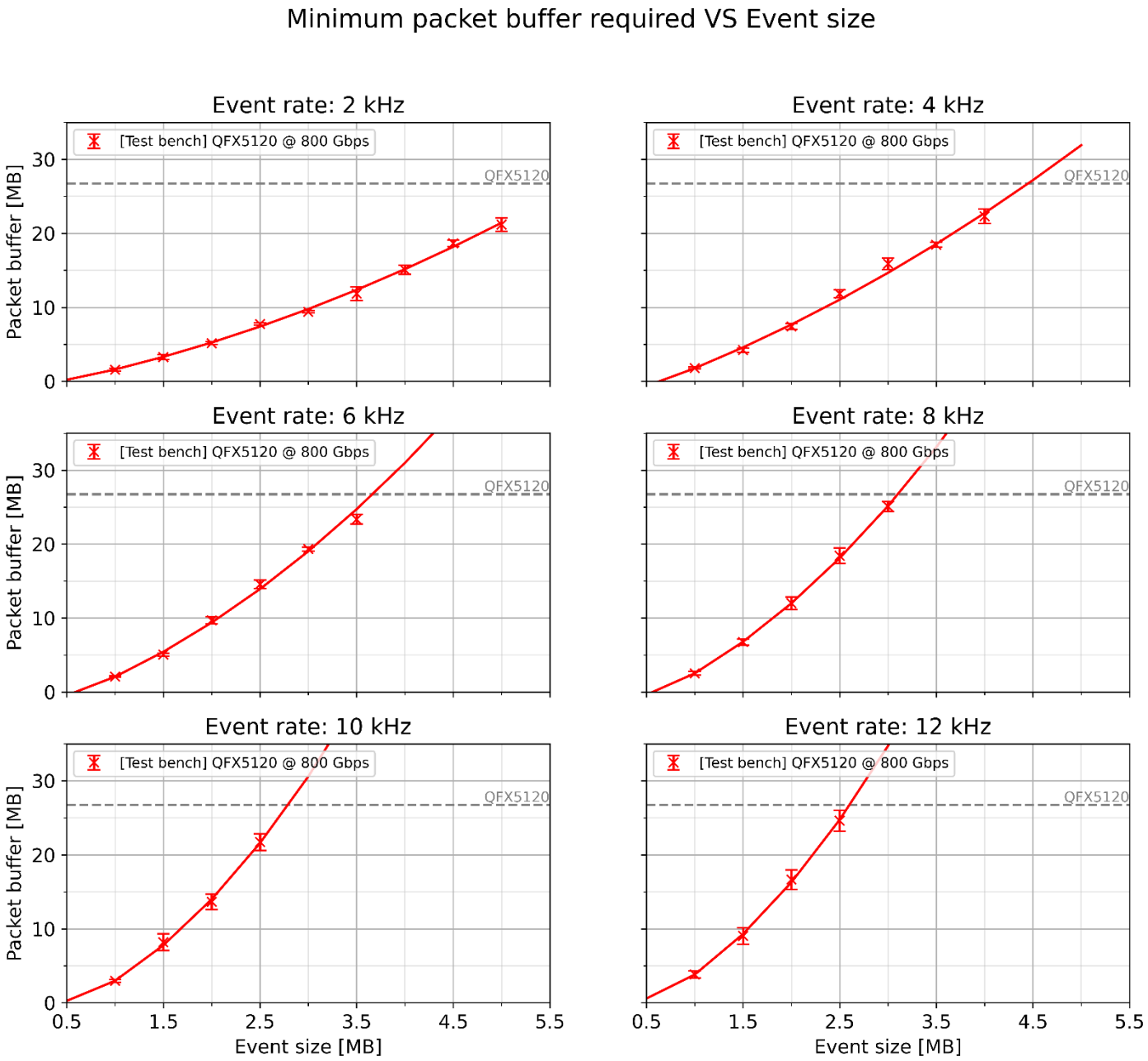
- Buffer pool: 27 MB shared + 5 MB fixed
- Installed BW: 8 x 100 Gbps uplink + 32 x 25 Gbps access

Buffer scan I
–
Event Size

Event size scan with the event rate fixed

- Progressively decrease the available buffer size until packet drops are observed
  - Measure the minimum packet buffer required
  - Repeat for many Event Rates: 2 to 20 kHz

Data fit well with a 2nd order polynomial

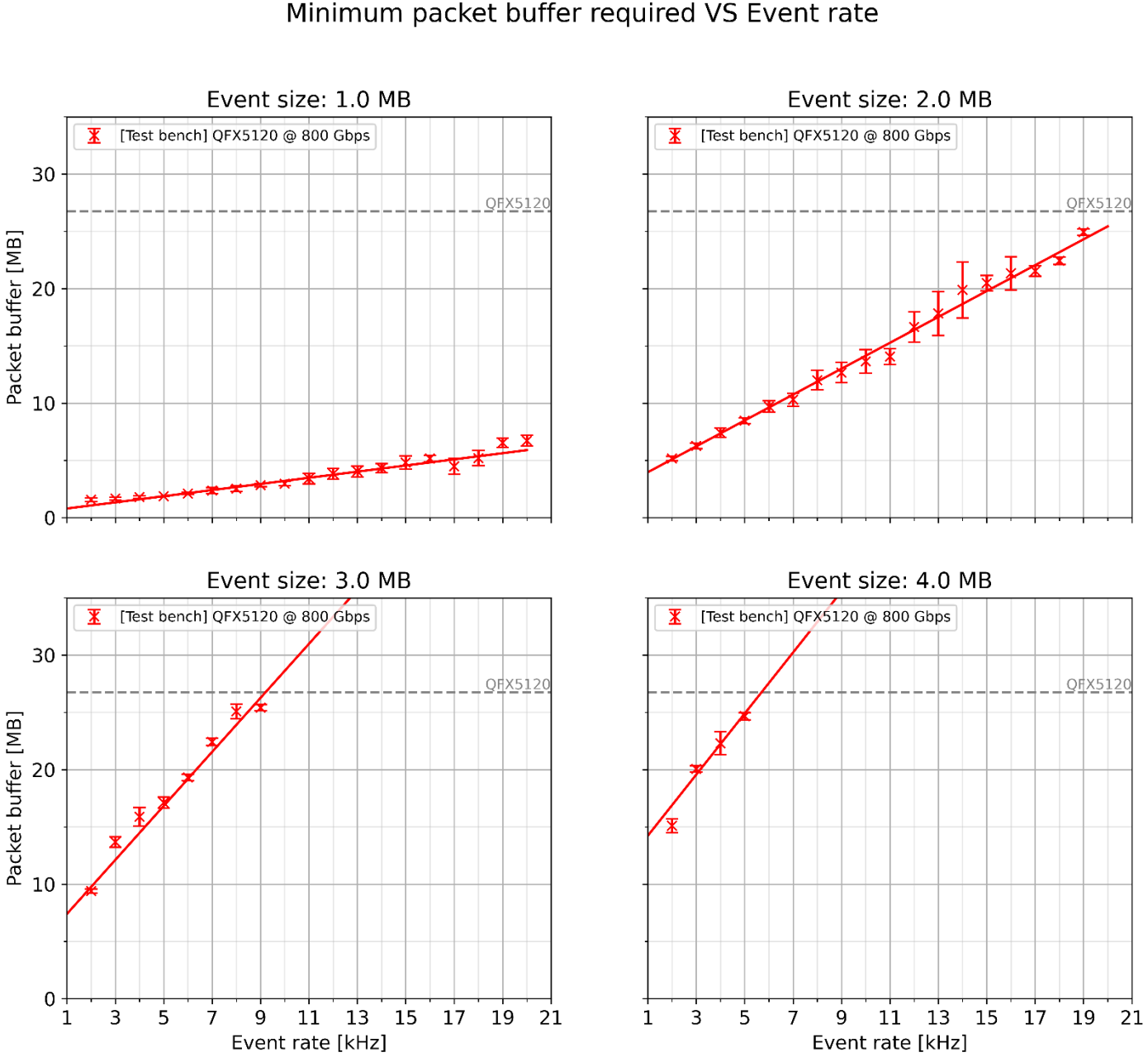


Buffer scan I
-
Event Rate

Event rate scan with the event size fixed

- Progressively decrease the available buffer size until packet drops are observed
  - Measure the minimum packet buffer required
  - Repeat for many Event Sizes: 1 to 5 MB

Data fit well with a 1st order polynomial



Buffer scan II
-
Analytic model

Simple analytic model for buffer occupancy

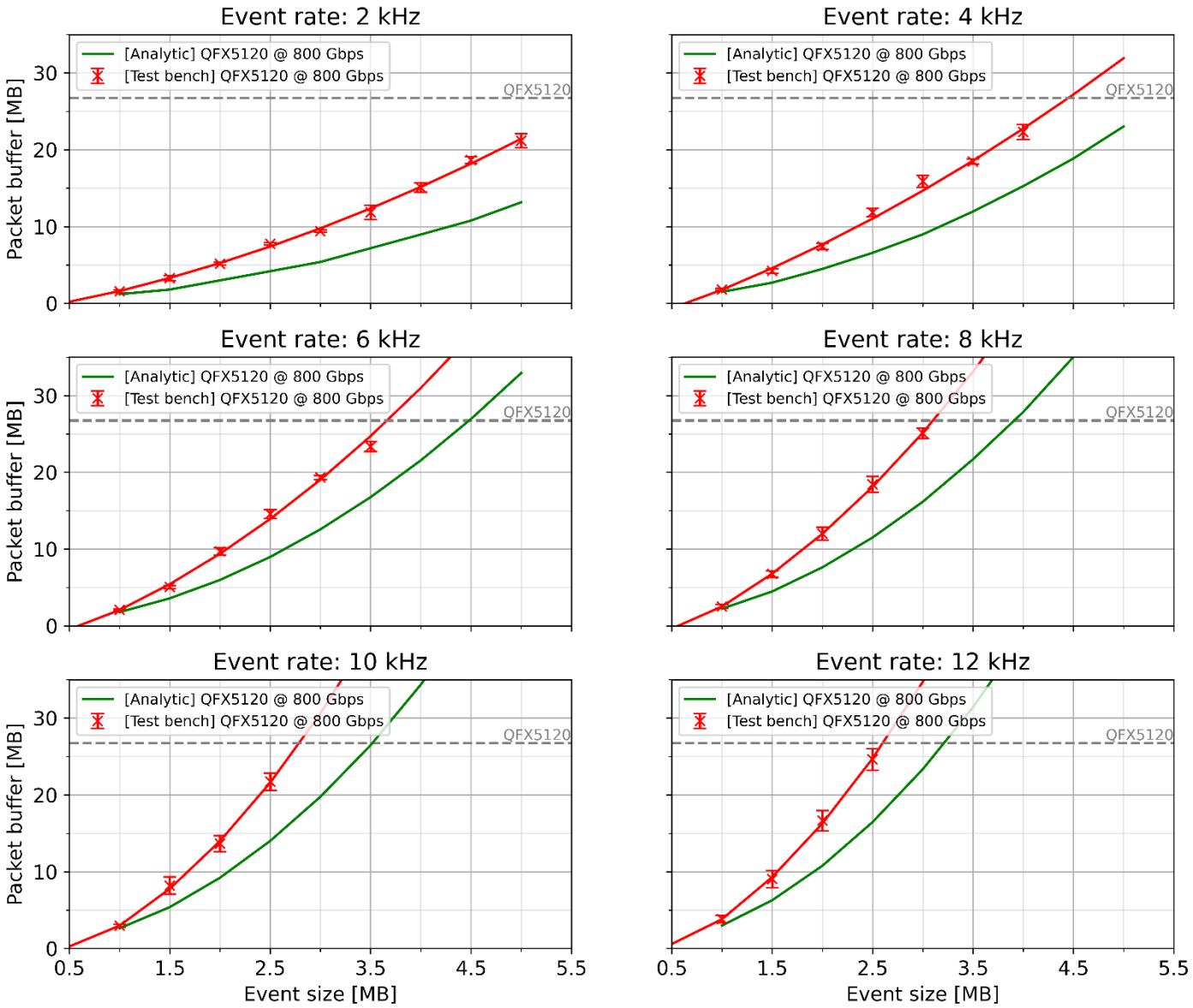
- Calculates the **peak** buffer occupancy
- **Not universal**: only for our DAQ traffic patterns

Not accurate but correctly captures the **quadratic** behavior

$$Buffer_{MAX} = N\left[\frac{I - O}{I}s - \frac{(N-1)\,O}{2f}\right]$$

$$with\ N = \left\lfloor \frac{s\,f}{O} \right\rfloor + 1\ where \begin{cases} I \equiv Input\ BW \\ O \equiv Ouput\ BW \\ s \equiv Event\ Size \\ f \equiv Event\ Rate \end{cases}$$

Minimum packet buffer required VS Event size



Buffer scan II
-
Analytic model

Simple analytic model for buffer occupancy

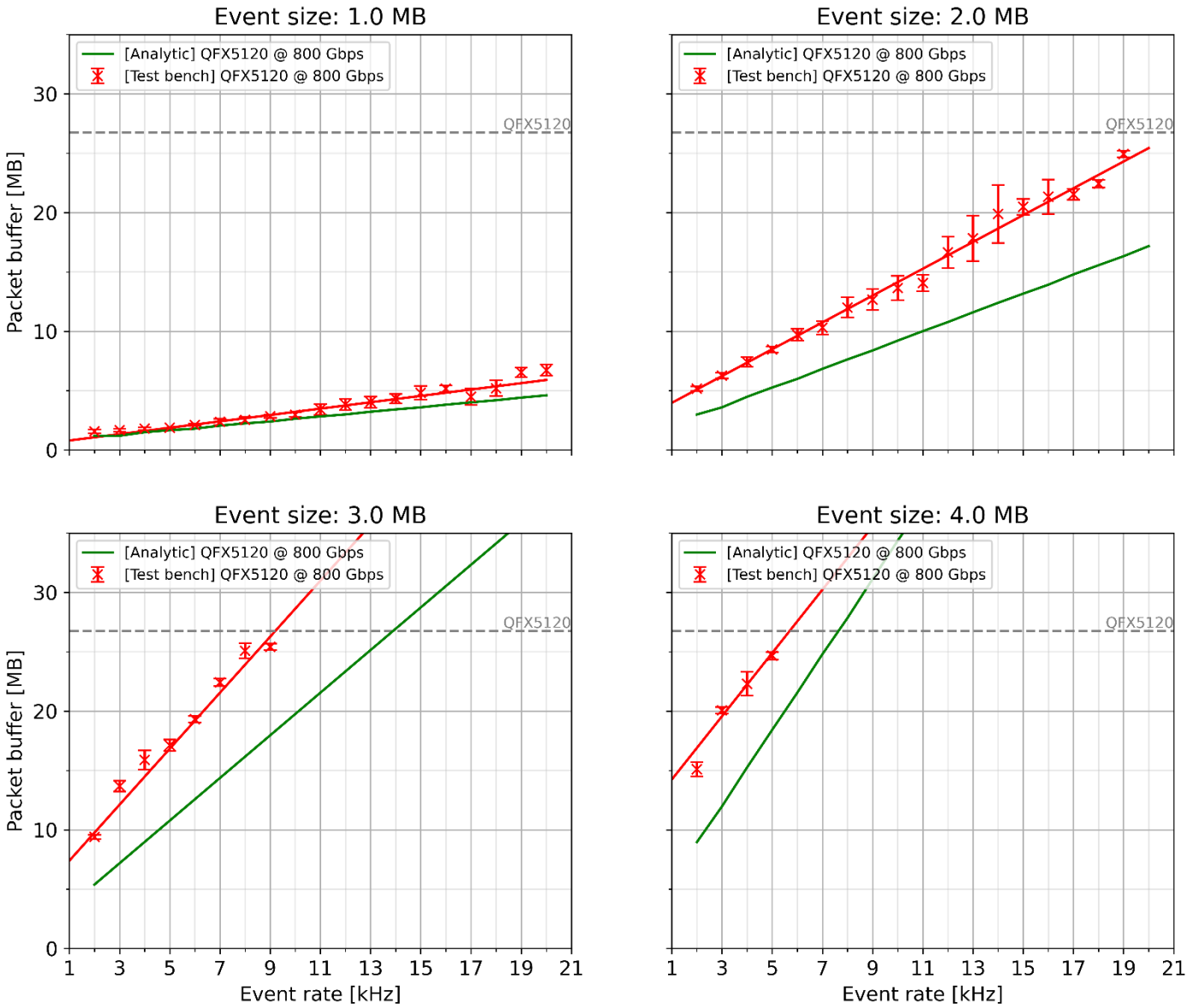
- Calculates the **peak** buffer occupancy
- **Not universal**: only for our DAQ traffic patterns

Not accurate but correctly captures the **linear** behavior

$$Buffer_{MAX} = N\left[\frac{I - O}{I}s - \frac{(N-1)\,O}{2f}\right]$$

$$with\ N = \left\lfloor \frac{s\,f}{O} \right\rfloor + 1\ where \begin{cases} I \equiv Input\ BW \\ O \equiv Ouput\ BW \\ s \equiv Event\ Size \\ f \equiv Event\ Rate \end{cases}$$

Minimum packet buffer required VS Event rate



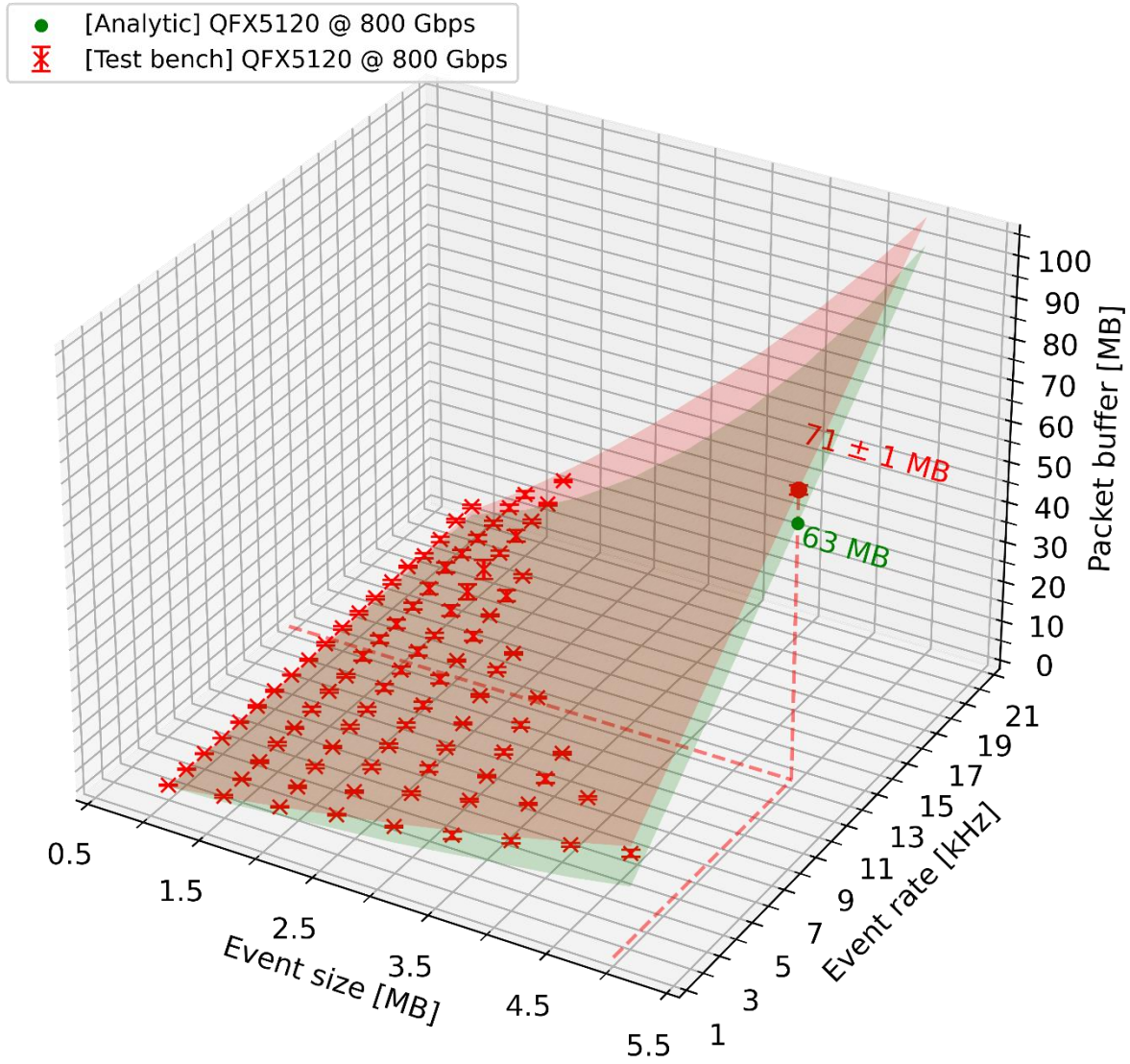
Building our model

- Our 2 dimensional dataset shows:
  - Quadratic behavior in one axis (Event Size: x)
  - Linear behavior on the other one (Event Rate: y)
- Fit with a generic polynomic surface:
  $F(x,y) = Ax^2y + Bx^2 + Cxy + Dx + Ey + F$
- Statistically significant fit ($R^2 = 0.99$) enables extrapolating to the target working point

Numerical extrapolation = $71 \pm 1\ MB$

Analytic extrapolation = $63\ MB$

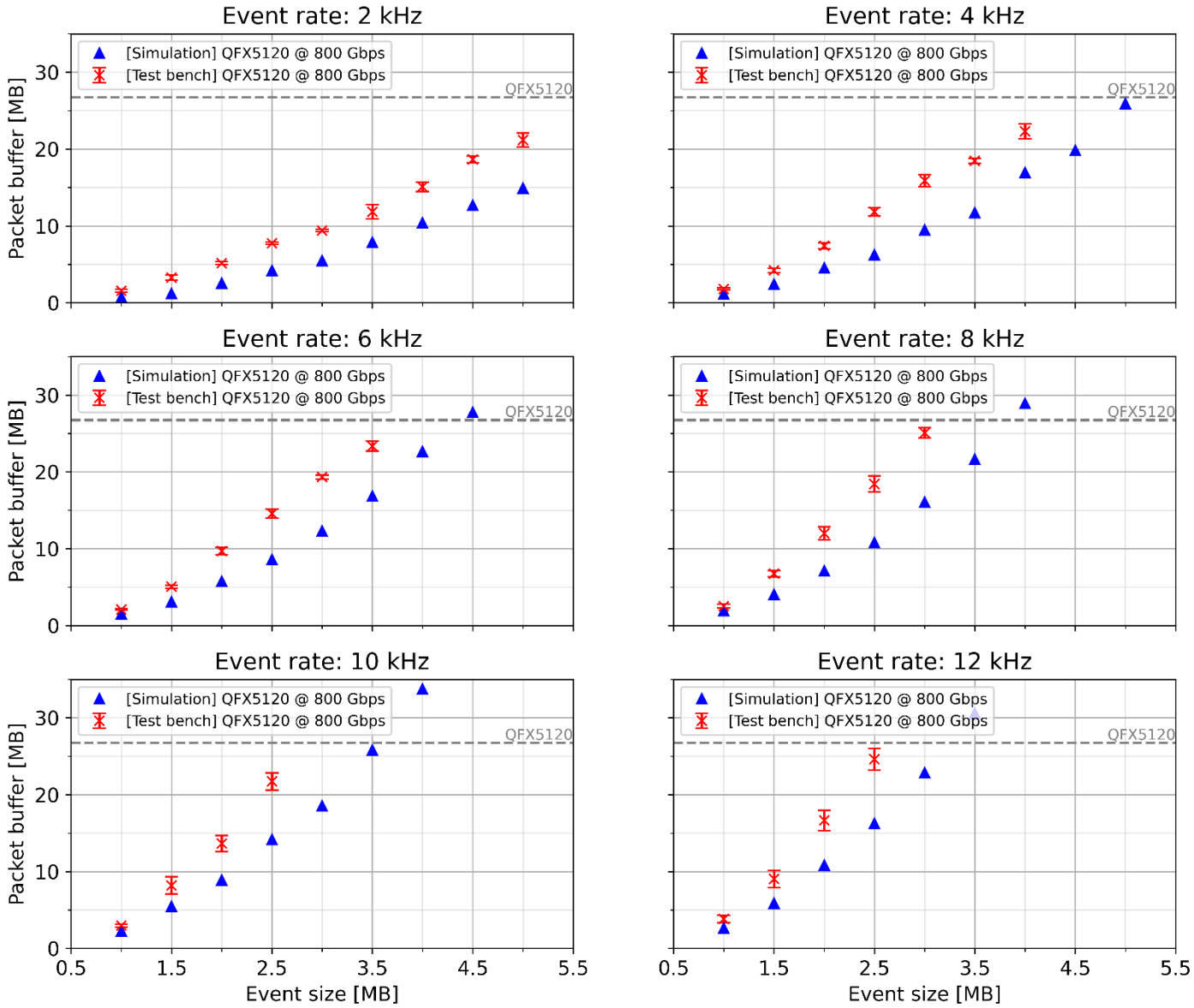
Minimum packet buffer required VS Event Size VS Event Rate



Buffer scan III - Simulation

- Discrete event simulation with TDAQ applications and network models
  - Key tool for scalability studies used since Run 2
- The model reproduces the shape of the measurements
  - Quadratic behavior for the Event Size

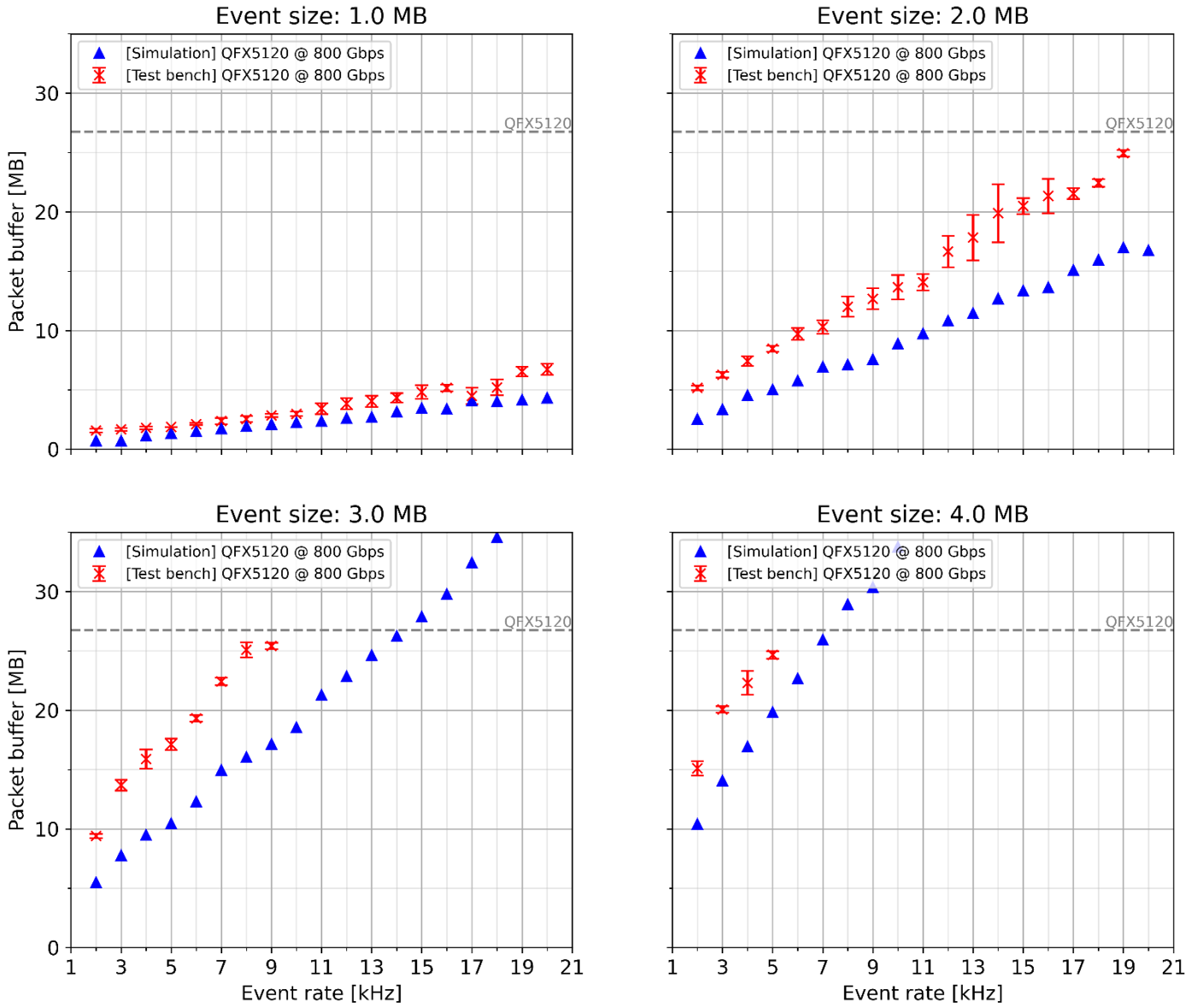
Minimum packet buffer required VS Event size



Buffer scan III - Simulation

- Discrete event simulation with TDAQ applications and network models
  - Key tool for scalability studies used since Run 2
- The model reproduces the shape of the measurements
  - Quadratic behavior for the Event Size
  - Linear behavior for the Event rate

Minimum packet buffer required VS Event rate

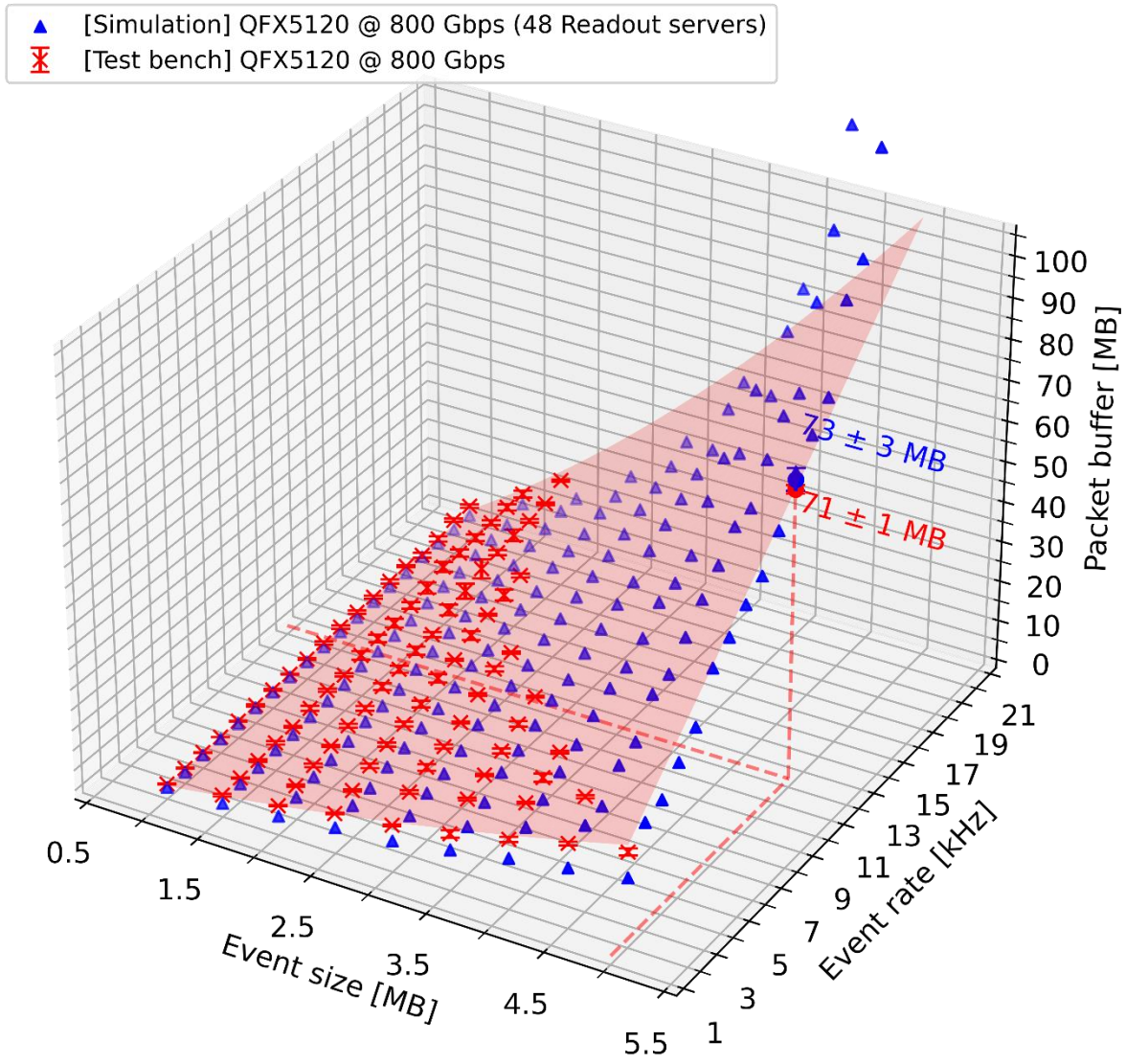


Buffer scan III
–
Simulation

- Discrete event simulation with TDAQ applications and network models
  - Key tool for scalability studies used since Run 2
- The model reproduces the shape of the measurements
  - Quadratic behavior for the Even Size
  - Linear behavior for the Even rate
- Simulated minimum buffer required lines up well with our previous estimates

Simulated buffer = 73 $\pm$ 3 MB

Minimum packet buffer required VS Event Size VS Event Rate

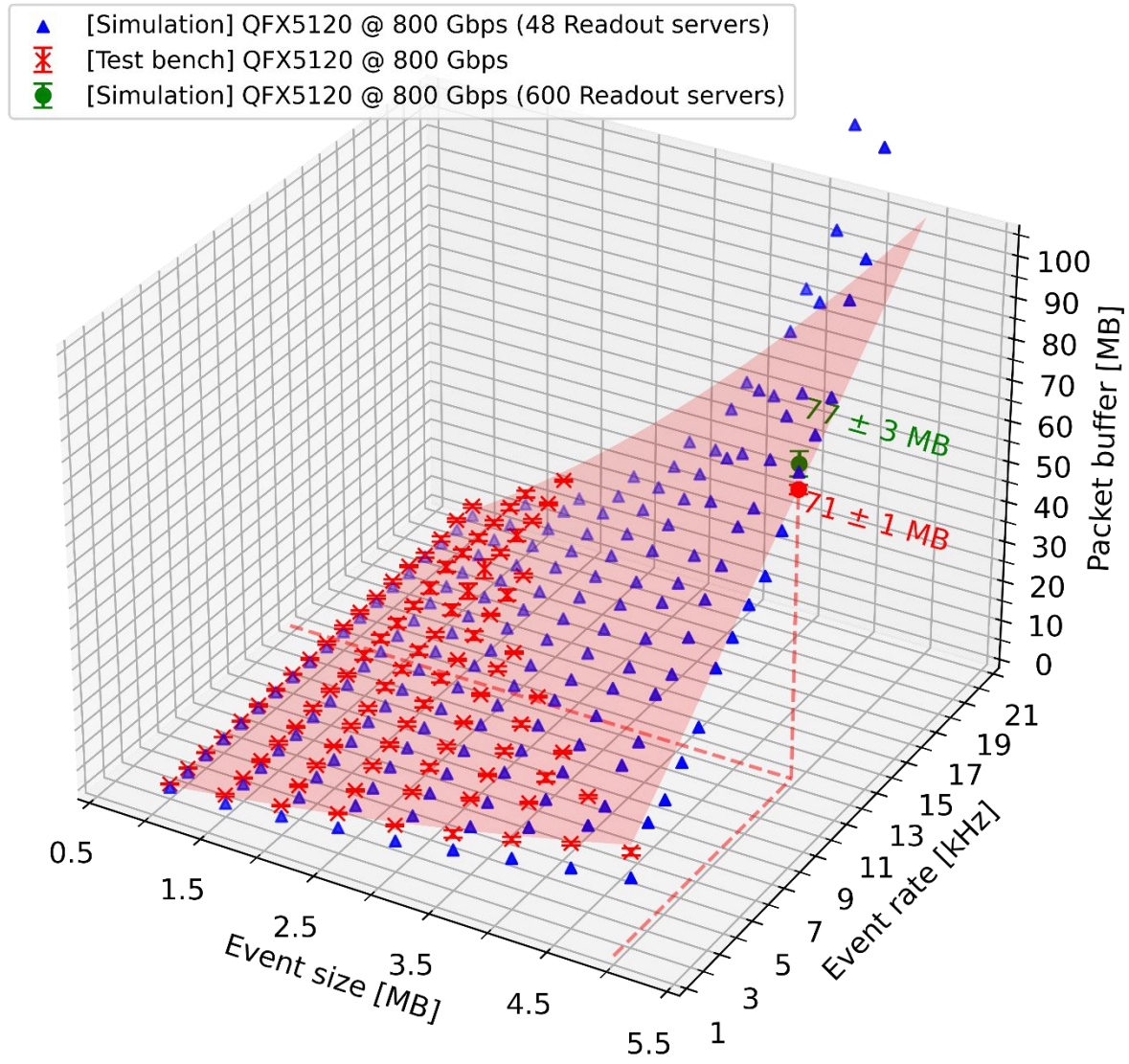


Buffer scan III
-
Simulation

- Discrete event simulation with TDAQ applications and network models
  - Key tool for scalability studies used since Run 2
- The model reproduces the shape of the measurements
- Can be used for **scalability** studies

Simulated buffer with 600 Readout servers (final system) = $77 \pm 3\ MB$

Minimum packet buffer required VS Event Size VS Event Rate



Conclusions and outlook

HL-LHC ATLAS TDAQ system: full Event Building with 5 MB events @ 1 MHz L0-trigger rate

- We built a scaled down prototype of the HL-LHC TDAQ system in a lab
  - Available buffer in a BCM Trident 3-based platform (QFX5120) was not enough to prevent packet drops
  - The problem is solved by increasing the available buffer size (e.g. ACX7100) -> x2.5 the cost
- We estimated buffer size requirements using 3 different methods
  - Findings can help define TDAQ network hardware for the HL-LHC system
- Now working on modifying the traffic patterns to avoid the *TCP incast*
  - Preliminary results very promising -> Could reach the target rate in a controlled environment
- Final network requirements will only be known when the EF system specifications are defined
  - A set of automated tools is now available to perform a quick re-evaluation
  - A validated simulation model ready to be used for design and scalability studies

Backup

TCP incast

TCP incast is a network congestion problem that can occur in data centres when multiple data sources simultaneously send data to a single destination.

TCP congestion control algorithm, regardless of the congestion window size, can't prevent the TCP incast when there is a sufficient number of data sources.