GOETHE UNIVERSITÄT FRANKFURT AM MAIN

High-throughput data distribution for CBM online computing

Jan de Cuveland
cuveland@compeng.uni-frankfurt.de

Dirk Hutter
hutter@compeng.uni-frankfurt.de

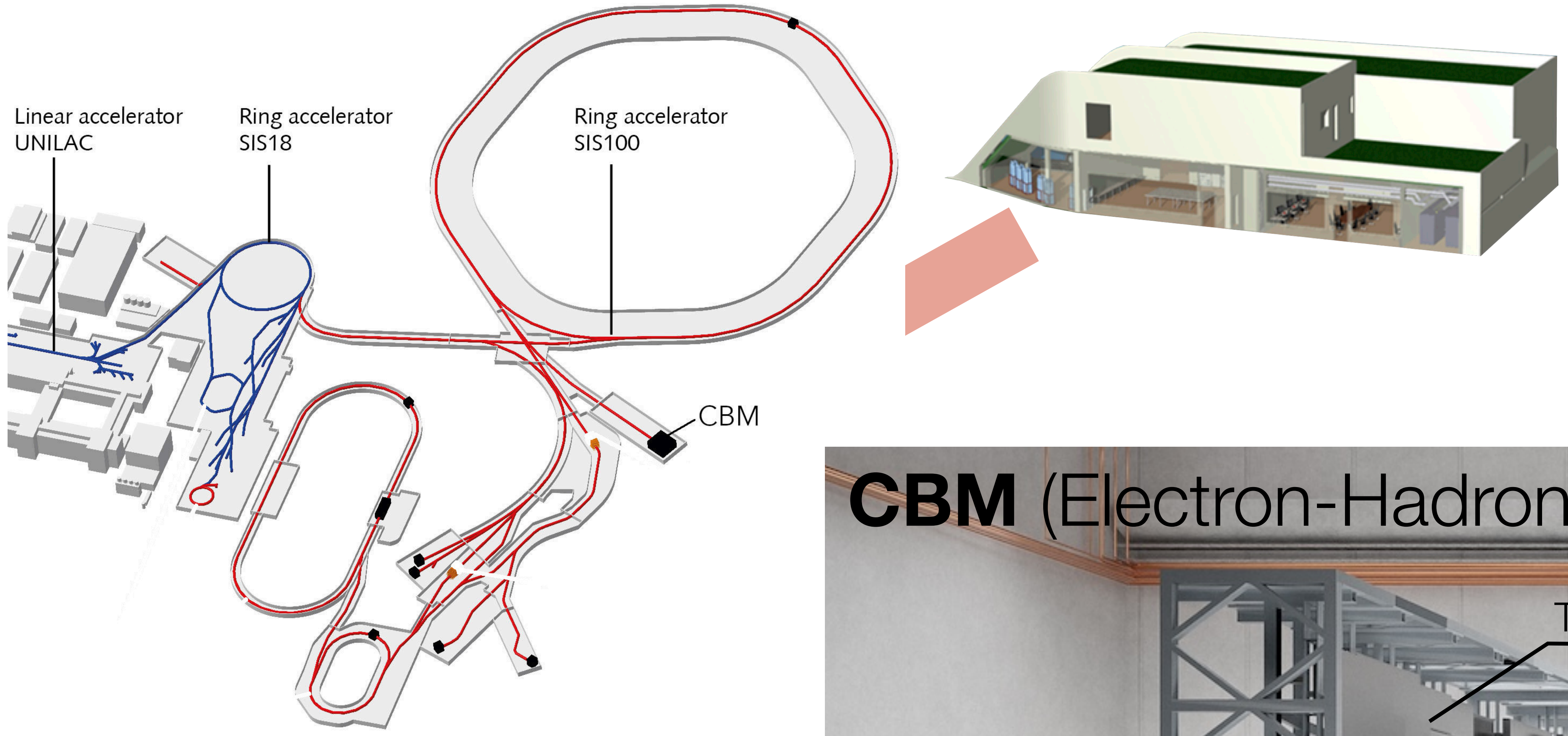
Prof. Dr. Volker Lindenstruth
FIAS Frankfurt Institute for Advanced Studies
Goethe-Universität Frankfurt am Main, Germany

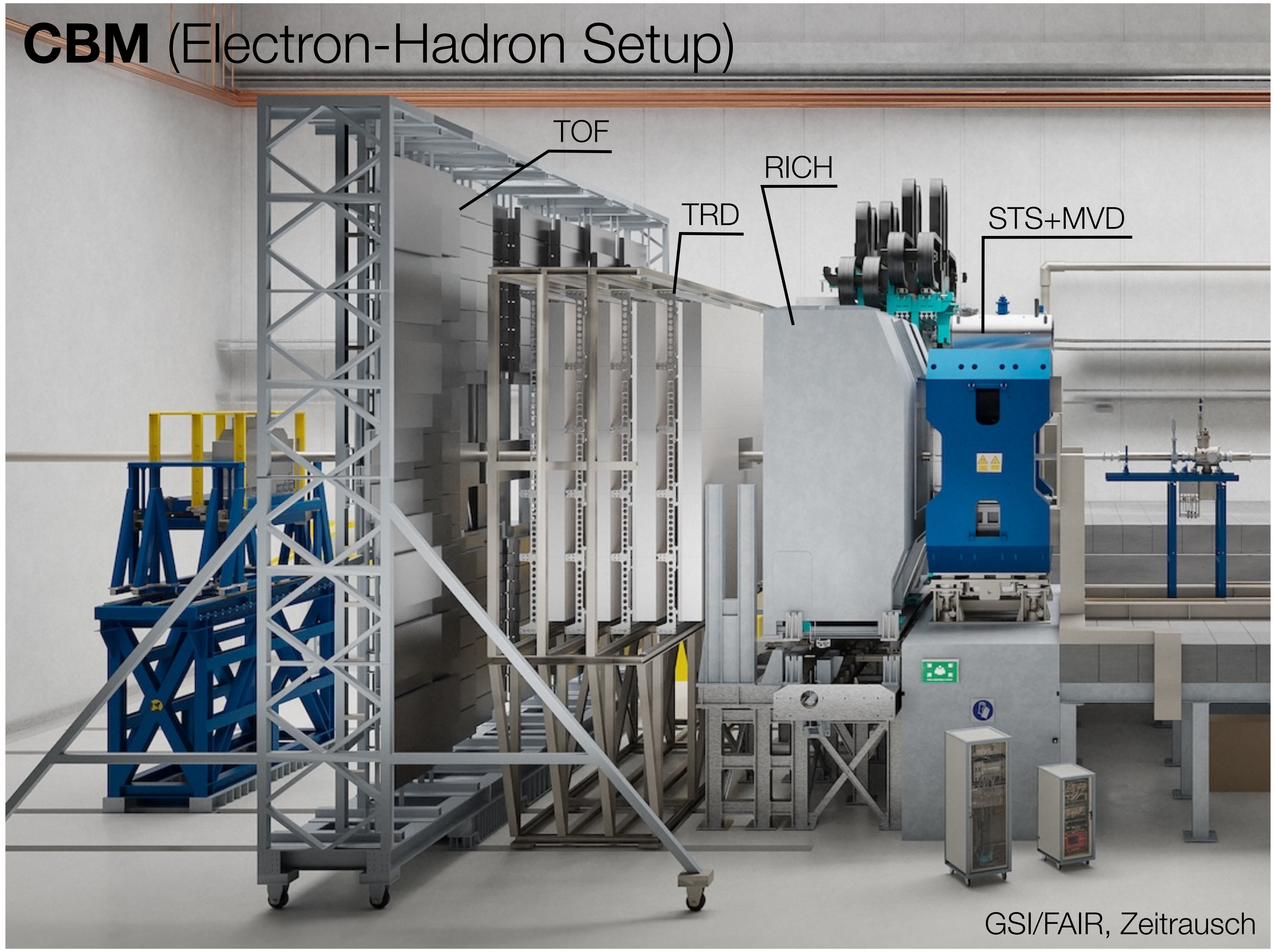
SPONSORED BY THE
Federal Ministry of Education and Research

CBM

Conference on Computing in High Energy and Nuclear Physics (CHEP) 2024, Track 2
2024-10-21 in Krakow, Poland

The CBM experiment at FAIR





- Fixed-target heavy-ion experiment at FAIR in Darmstadt
  - Physics goal: exploration of the QCD phase diagram
  - Plan: ready for beam in 2028
- High interaction rates of up to **10 MHz** and up to 700 charged particles in aperture
- Complex (topological) trigger signatures
- Full **online event reconstruction** needed

→ Self-triggering free-streaming readout electronics

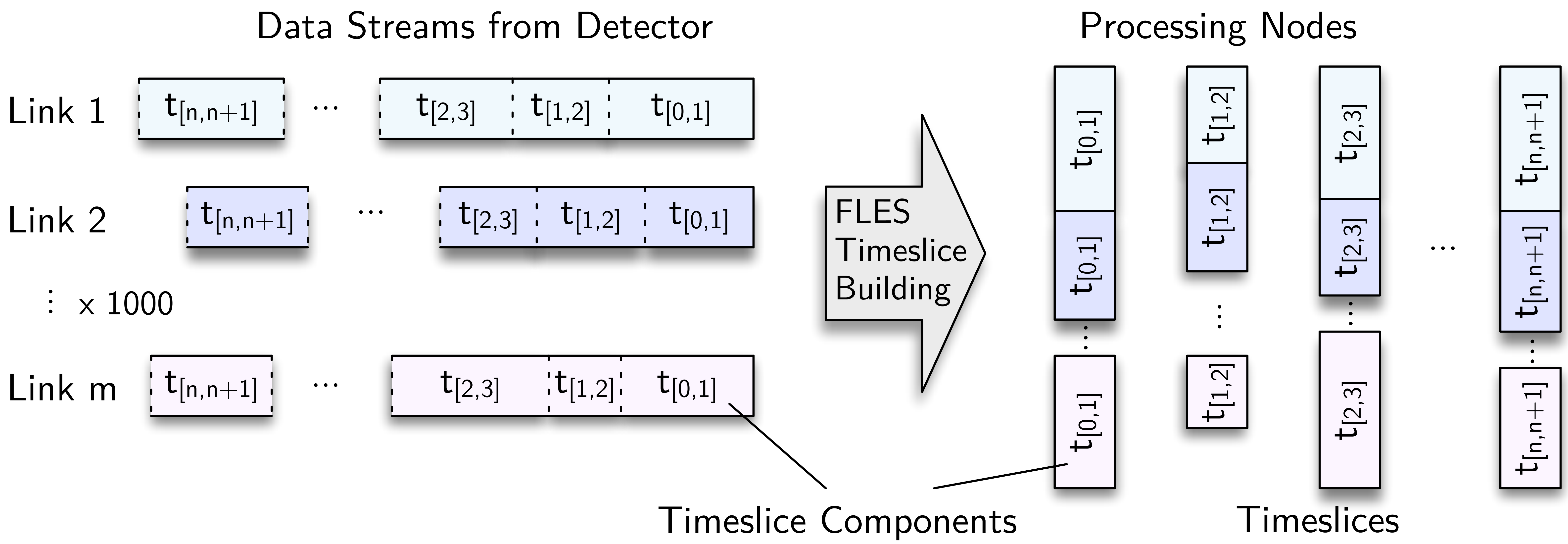
→ Event selection exclusively done in an HPC cluster (FLES)

Timeslice data model

- **CBM particularity:** Unlike in a collider, there is no a-priori knowledge when a collision happens: no bunch crossing, quasi-continuous beam on target. Events have to be defined from the data stream in software.

- **Timeslice:** a collection of raw (packed) data from all detector systems within a fixed time interval. Typical size: several GB; contains data from several thousand collisions.

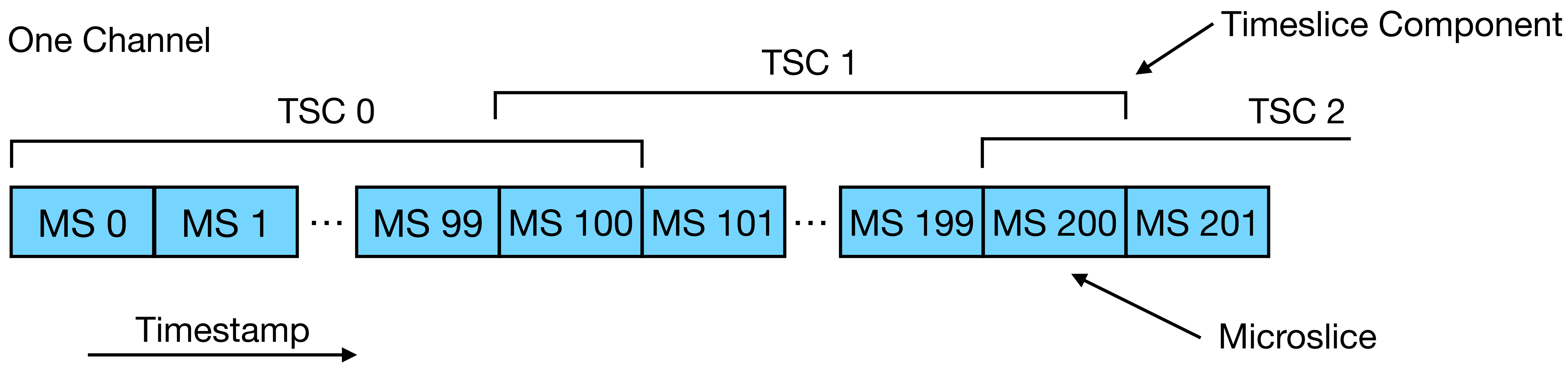
- **Flesnet:** a software package that assembles timeslices from the incoming data streams.



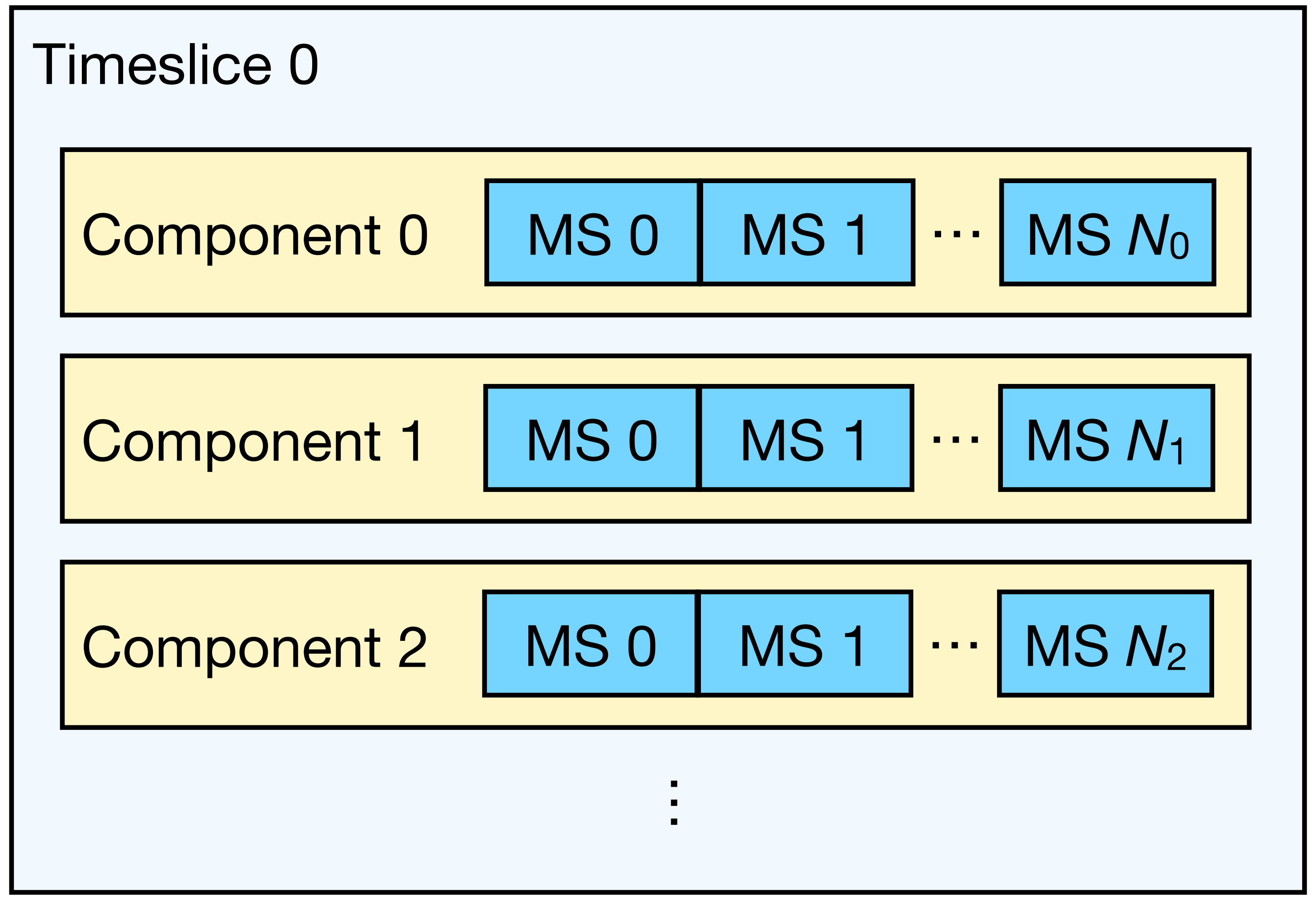
- Timeslice data management concept
  - A timeslice is **self-contained** and can be **analyzed independently**
  - Distribute different timeslices to **different processing nodes**
  - Subsequent timeslices **overlap** to handle data at boundaries
    - Guarantee: All measurements with event time in core interval are included.
    - Use COG in time of reconstructed event to avoid duplicate detection

Microslice-based timeslice building

1. Partition the detector message streams into short, context-free time intervals: **microslices**
   - Built by detector-specific FPGA design. Example: ~100 µs in experiment time
2. Combine subsequent microslices to one timeslice component (TSC)
   - Include overlap as configured



3. Combine timeslice components from all sources to processing intervals: **timeslices**



Benefits

- Decouples online data management and detector data format
- Allows timeslice **overlap**
- Allows easy **parallel processing** of local reconstruction
- Allows **efficient** zero copy timeslice building

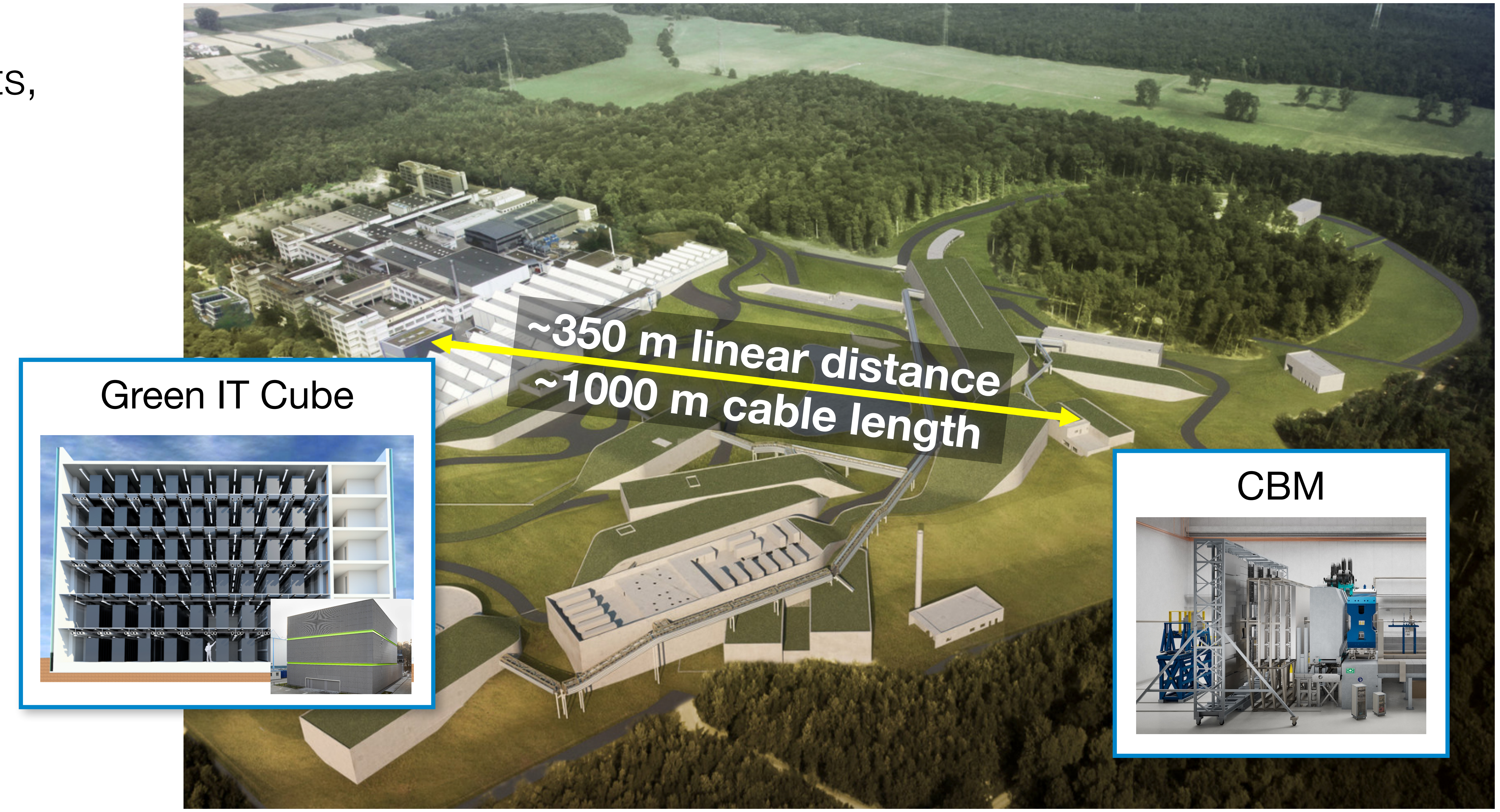
CBM first-level event selector (FLES)

- **CBM particularity:** Unlike in the LHC experiments, for example, the online compute nodes do not belong to CBM, but are provided by FAIR-IT as shared resources (guaranteed during run times).

- **Dual-cluster HPC system**
  - Commodity PC hardware
  - Design input data rate > 1 TByte/s

- **Part 1: Entry node cluster**
  - Located in the CBM service building
  - FPGA-based custom PCIe input interface
  - **Exclusive** to CBM

- **Part 2: Shared compute cluster**
  - Located in the Green IT Cube data center



Consequences
- Transmit 1 TByte/s over 1000 m distance
- Boundary condition for online computing architecture

CBM FLES online architecture

CBM detectors
- **10^7 events/s**
- Self-triggering front-end
- Data push architecture
- Time-stamped message streams

FLES input
- FPGA-based PCIe board
- Preprocessing and indexing
- High-throughput DMA engine

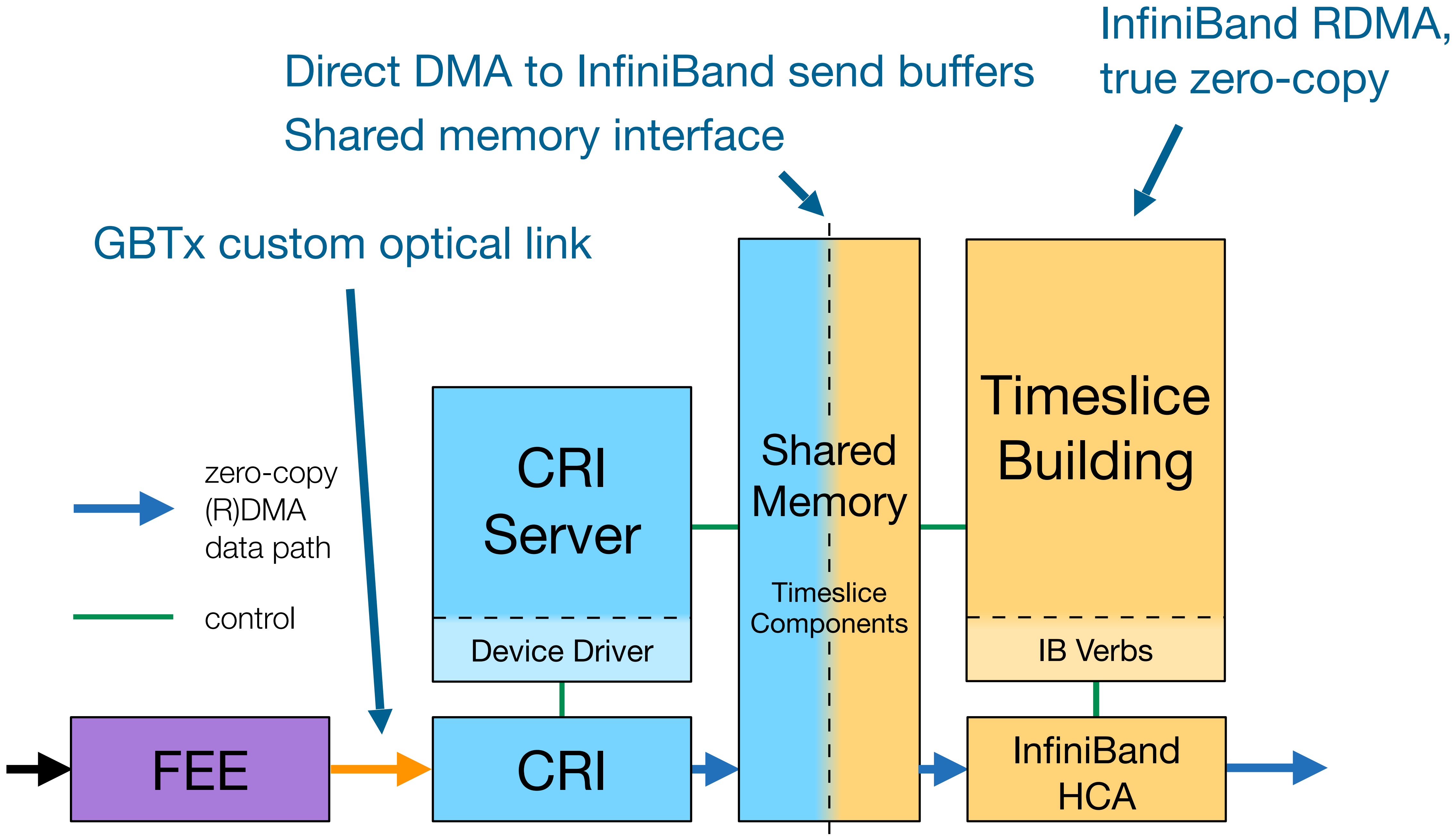
High-throughput timeslice building
- Up to 1 TByte/s input data rate
- RDMA-enabled network (InfiniBand)
- Long-range links

Online event selection
- ~ 60.000 cores
- Fast, vectorized many-core track reconstruction algorithms
- Identification of leptons and hadrons
- High-precision vertex reconstruction
- Complex global triggers

FLES input interface

Presentation later today by **D. Hutter** (Track 2)

- FPGA-based PCIe board: CRI
  - Prepares and indexes data for timeslice building
  - Custom PCIe **DMA interface**, full offload engine
- Optimized data scheme for zero-copy timeslice building
  - Transmit microslices via PCIe/DMA directly to **userspace buffers**
  - Buffer placed in Posix shared memory, can be registered in parallel for **InfiniBand RDMA**

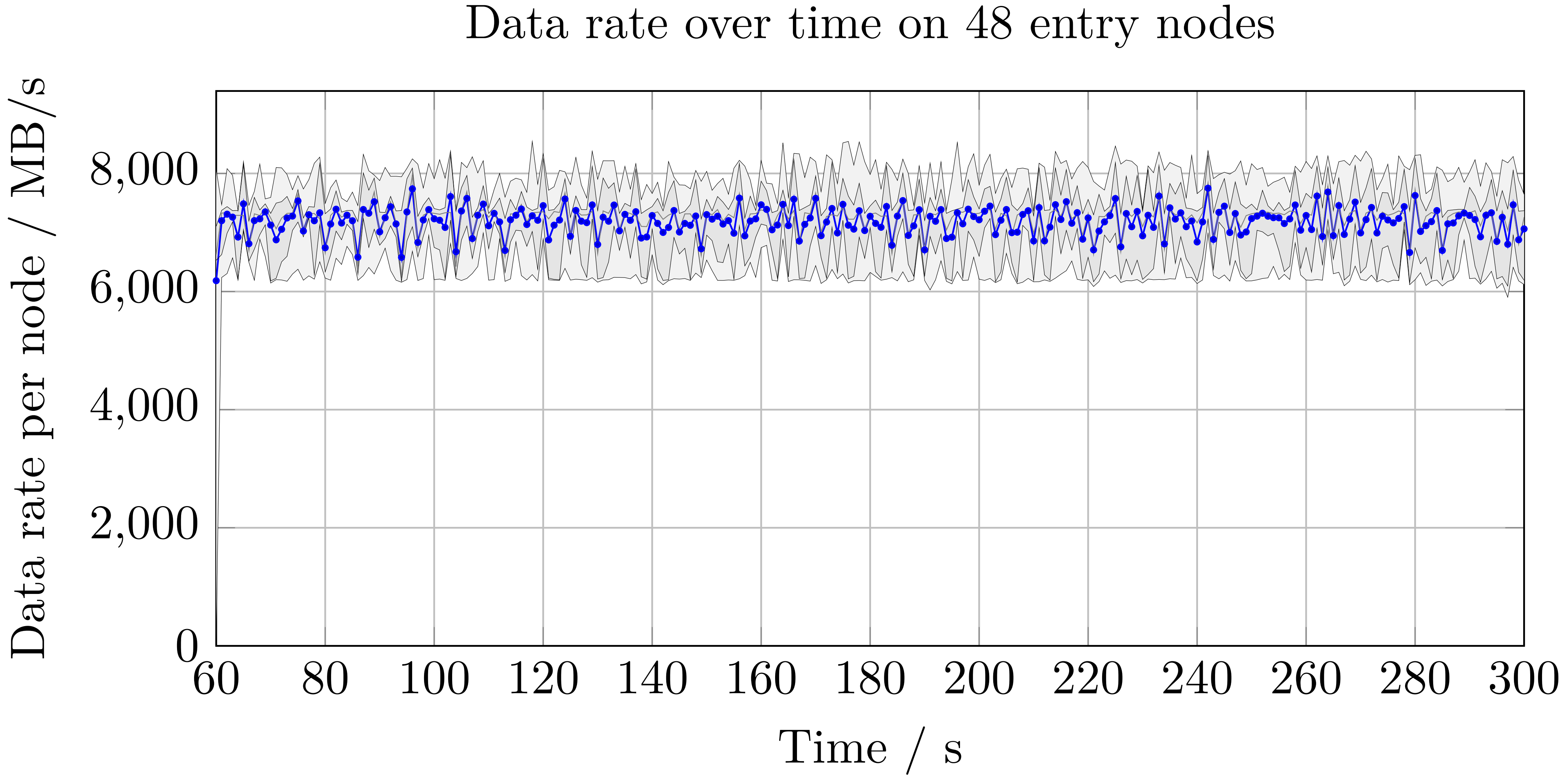


The CBM online data path

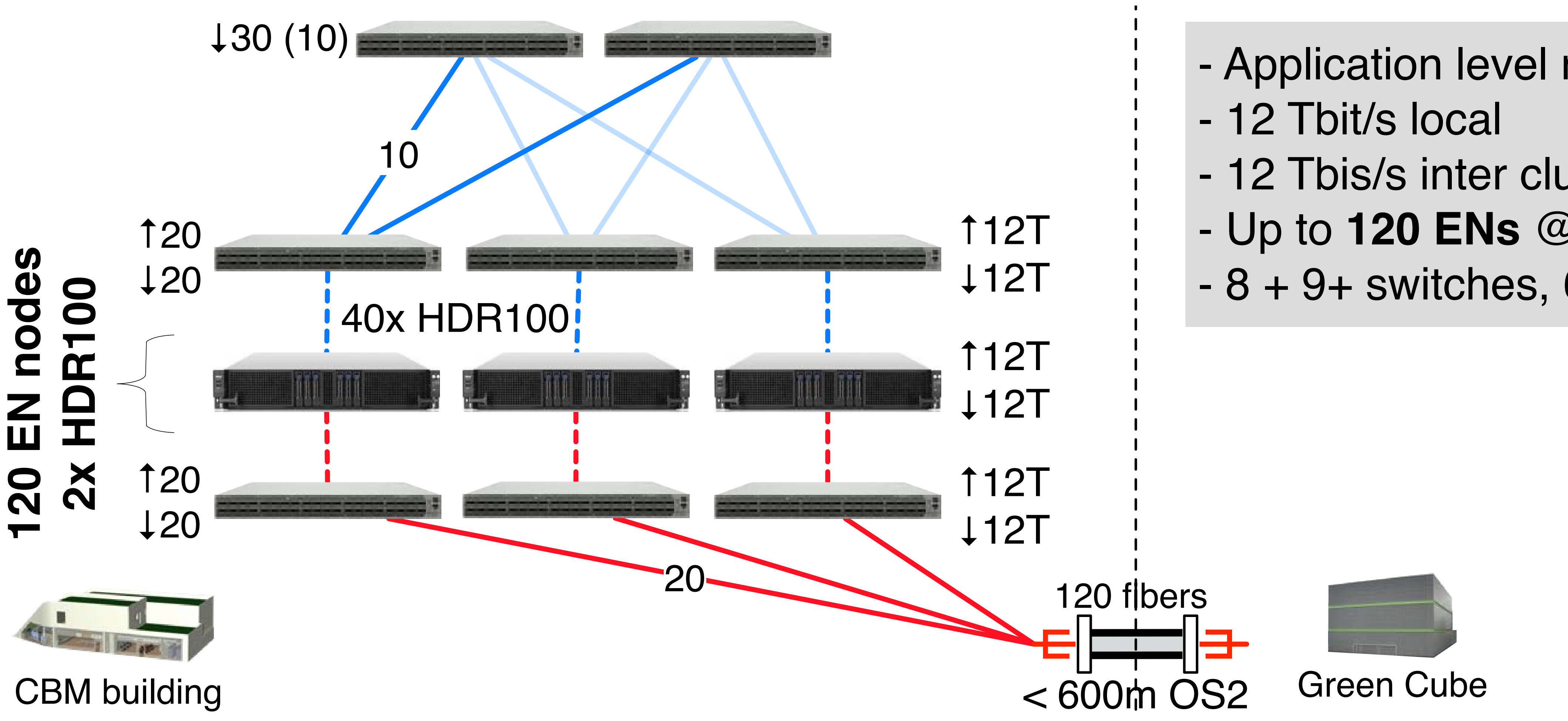
- RDMA-based timeslice building (Flesnet)
- Delivers fully built timeslice to reconstruction code

- Initial implementation of all components available
  - C++, Boost, IB verbs
  - Critical network performance optimized for > 1 TB/s

FLES network

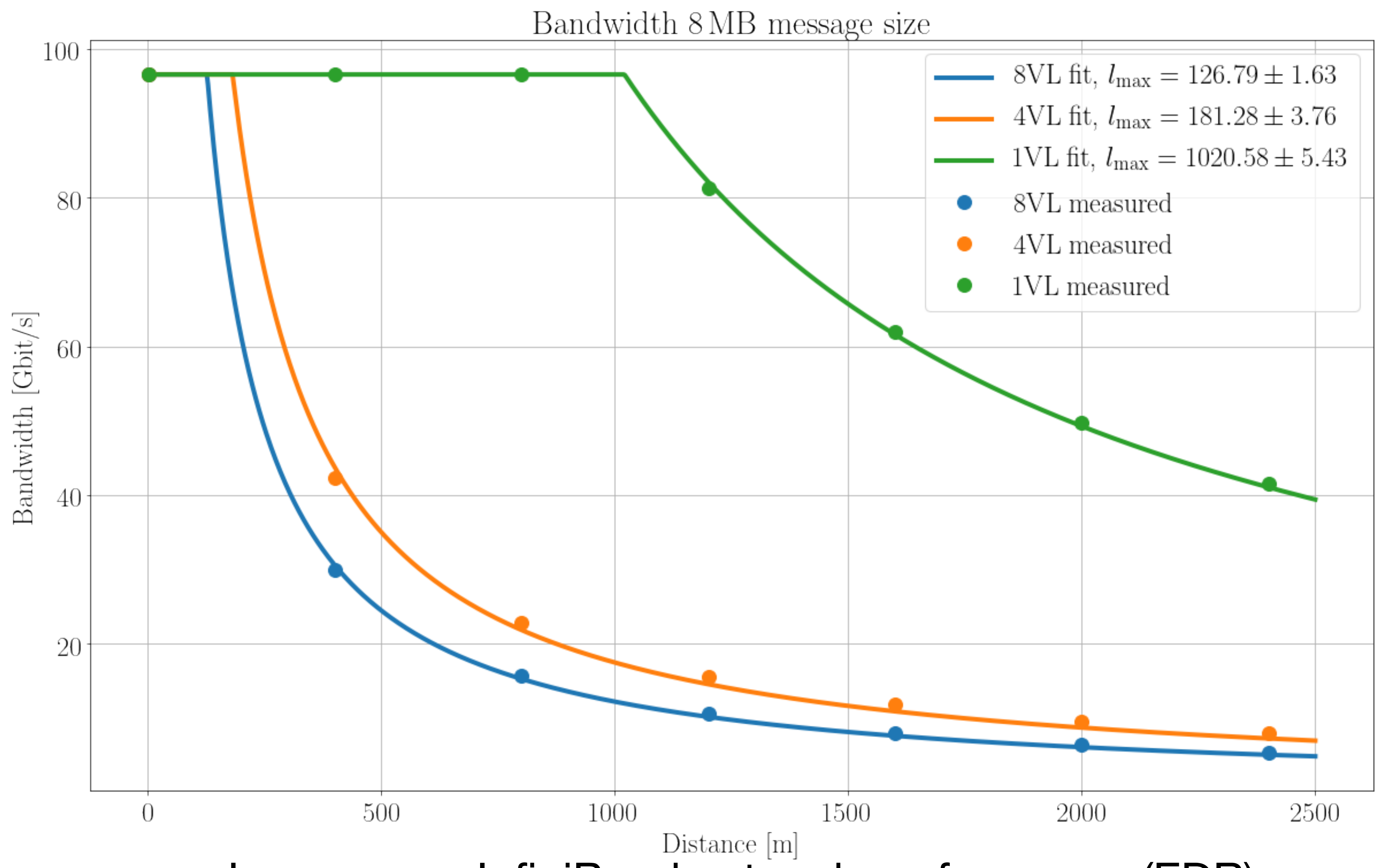


Test on Cray XC40 system (48 entry nodes, 48 build nodes)



- Application level routing
- 12 Tbit/s local
- 12 Tbis/s inter cluster
- Up to **120 ENs** @ 100 Gb/s in
- 8 + 9+ switches, 60 + 60 cables

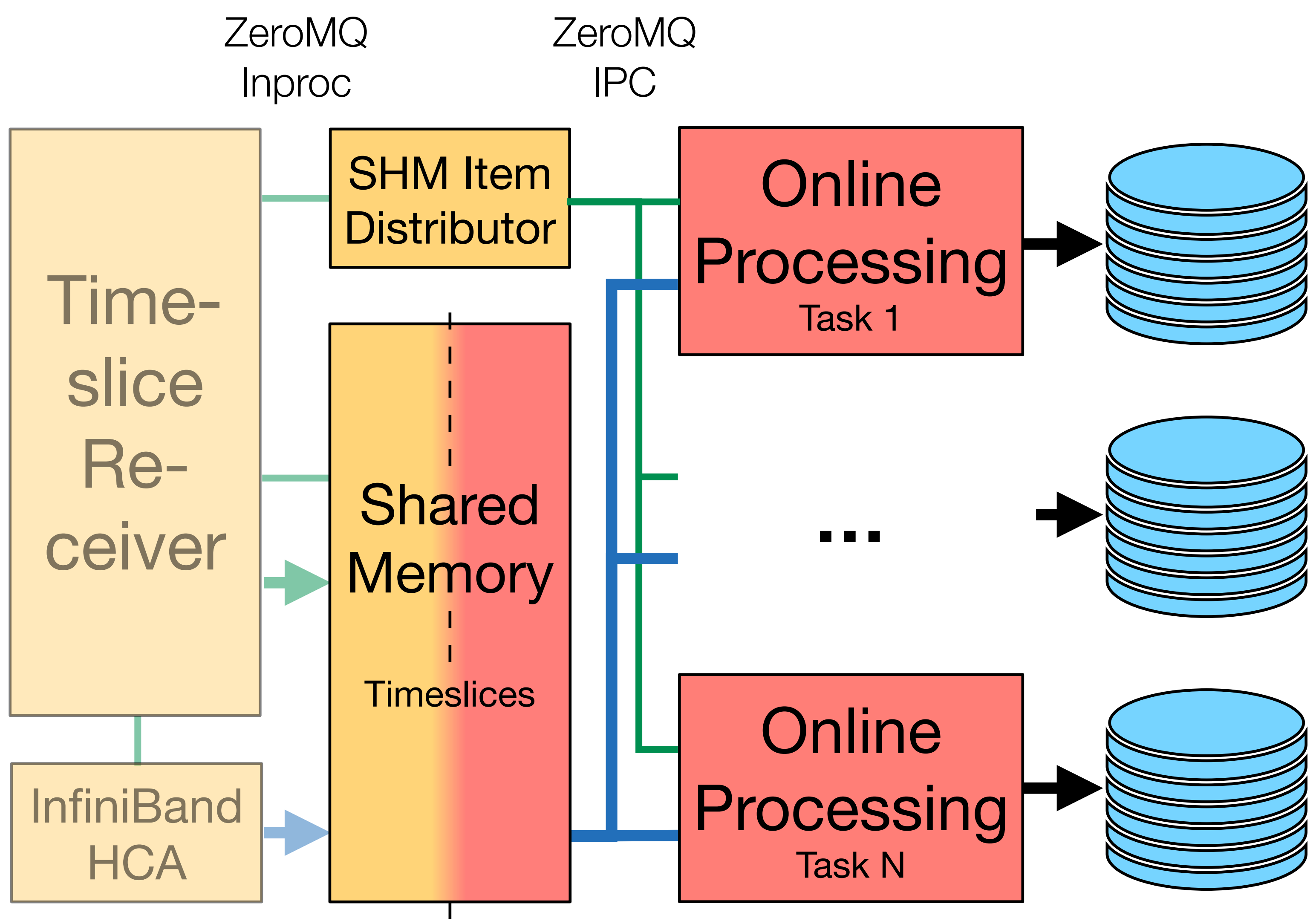
- Standard routing pattern suboptimal for **continuous all-to-all communication**
- **Optimized routing scheme** using RDMA leads to excellent performance (**>5 GB/s per node**)
- Measured approx. 345 GB/s sustained rate on 48 nodes
- Improve **long-range InfiniBand performance** using standard components by collapsing virtual lanes



Long-range InfiniBand network performance (EDR)

Online interface to timeslice data

- **Common problem:** different consumers need efficient access to data items on a node (here: timeslices)
- **Solution:** shared memory for data, managed by a dedicated distributor task → shm_ipc library
- Features
  - Queueing and reference-counting
  - Independent consumer processes with individual queueing schemes
    - With/without back pressure; subsampling; consumer groups
  - Implementation: Posix shared memory for the data and ZeroMQ messaging for arbitration
  - Full flexibility in starting and stopping consumers
- Used as a flexible online interface to CBM timeslices
  - Accommodates online data analysis, QA tasks, raw data storage, …



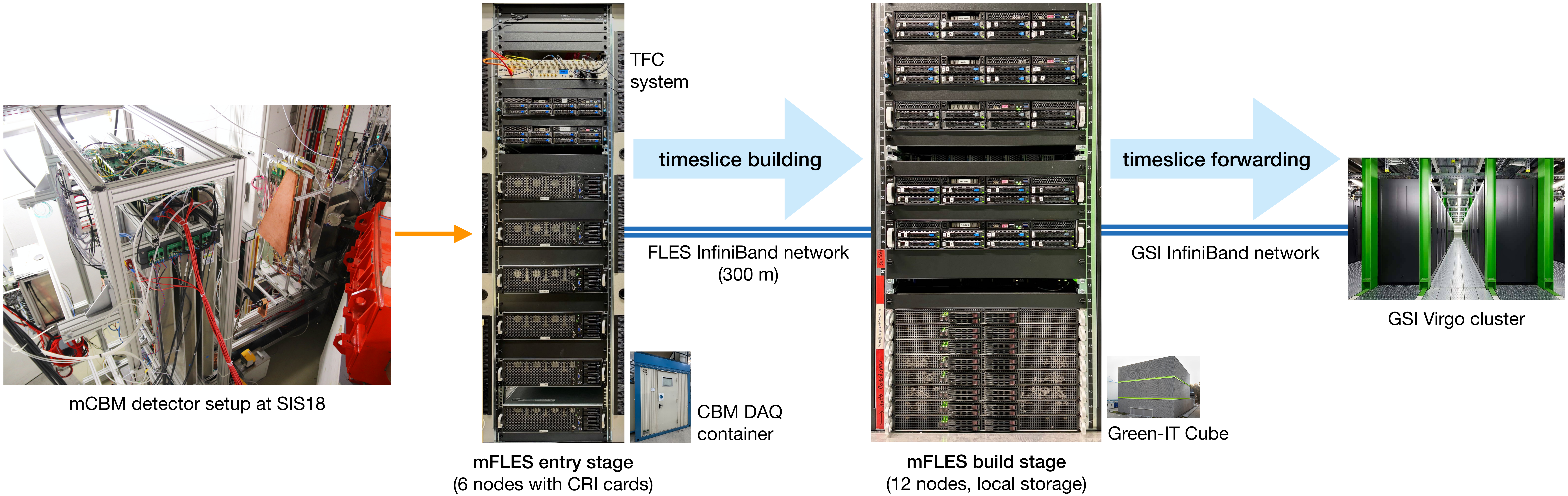
Example queueing schemes

QueueAll: All item are queued and eventually delivered; can create back pressure

PrebufferOne: Opportunistic delivery; keeps consumer busy but may skip items

Skip: Always wait for the newest item, do not queue; may skip items

Full-system test at mCBM

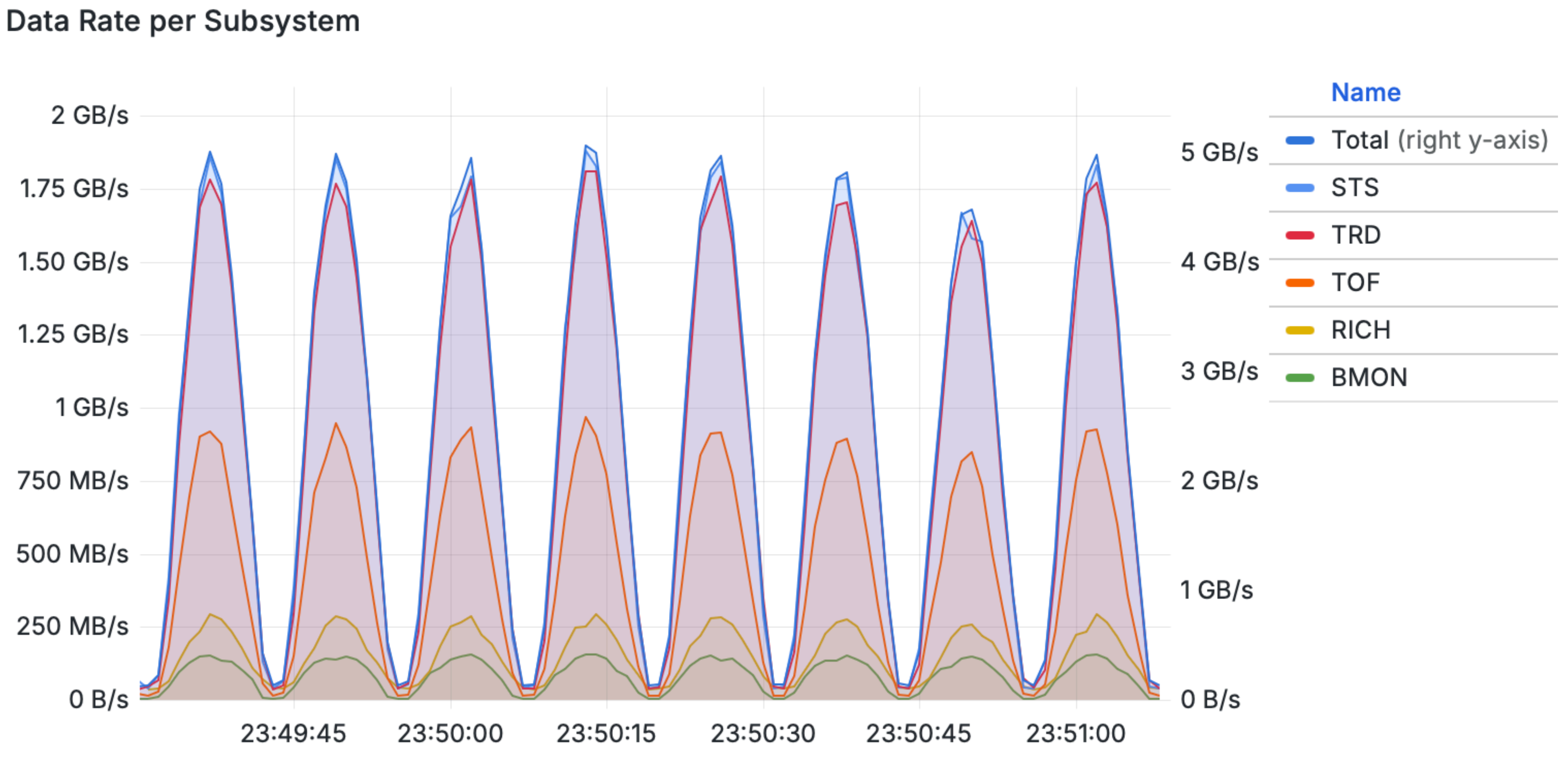


- FAIR Phase-0 experiment mCBM:
  - A complete slice of the full CBM system (hardware+software)
  - Apply detectors and event selection to live physics data
  - Study integration (and identify missing pieces) in a full-system test
  - Regular data taking campaigns

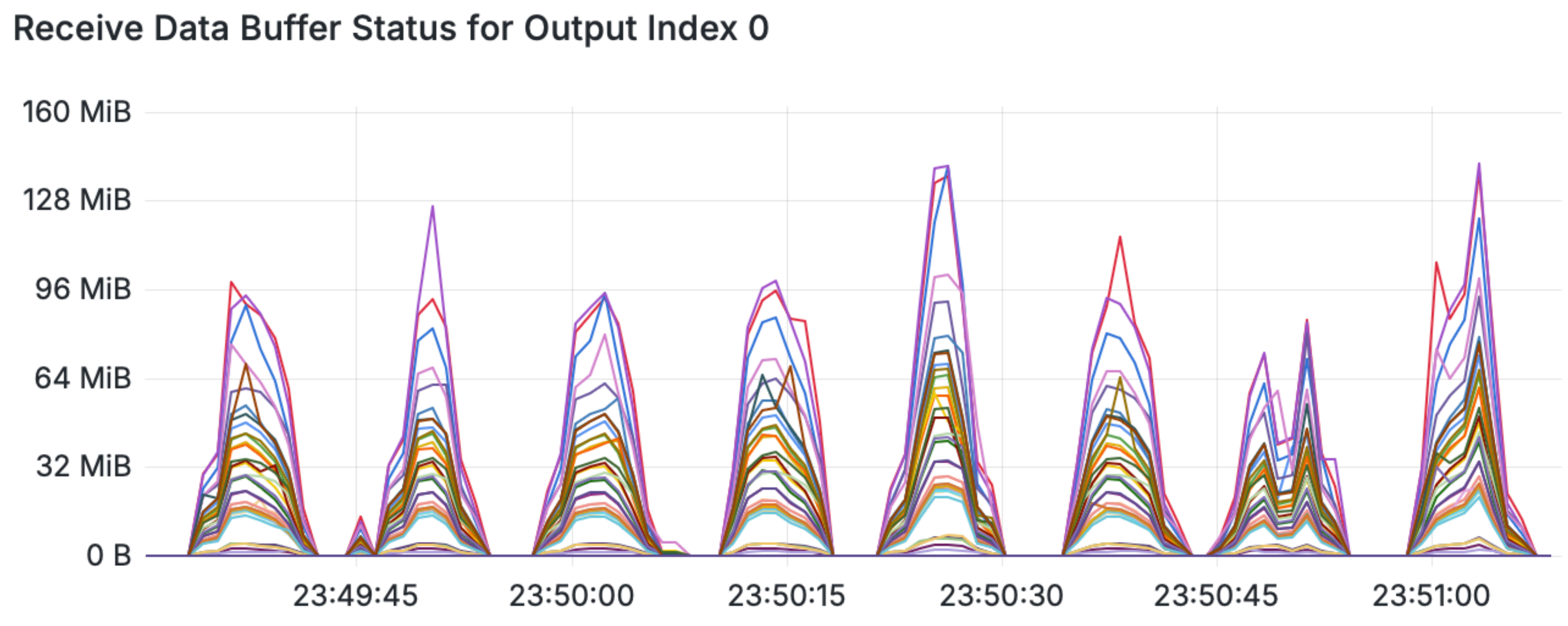
- mFLES:
  - mFLES cluster with CRIs and FLES software is the **central data taking system**
  - Demonstrator and development platform for FLES software
  - Setup includes **all key components** needed for CBM@SIS100
    - Hardware currently approx. 2 % of foreseen FLES system

Full-system test at mCBM

- FLES control and monitoring system
  - Automated run control with **configuration and process management** on mFLES cluster
  - Successful productive operation of full FLES/DAQ chain from CRI to timeslices
  - **Online monitoring** of all critical parameters
- Example: May 2024 mCBM campaign
  - 5 detector systems: STS, TOF, RICH, TRD, BMON
  - Distributed data taking:
    4 entry nodes, 4 build nodes, 44 components
  - Peak data rates above **5 GByte/s**
  - Full Flesnet software chain with **timeslice building** and **online processing** using multiple timeslice consumers



Run 3035 on 2024-05-10



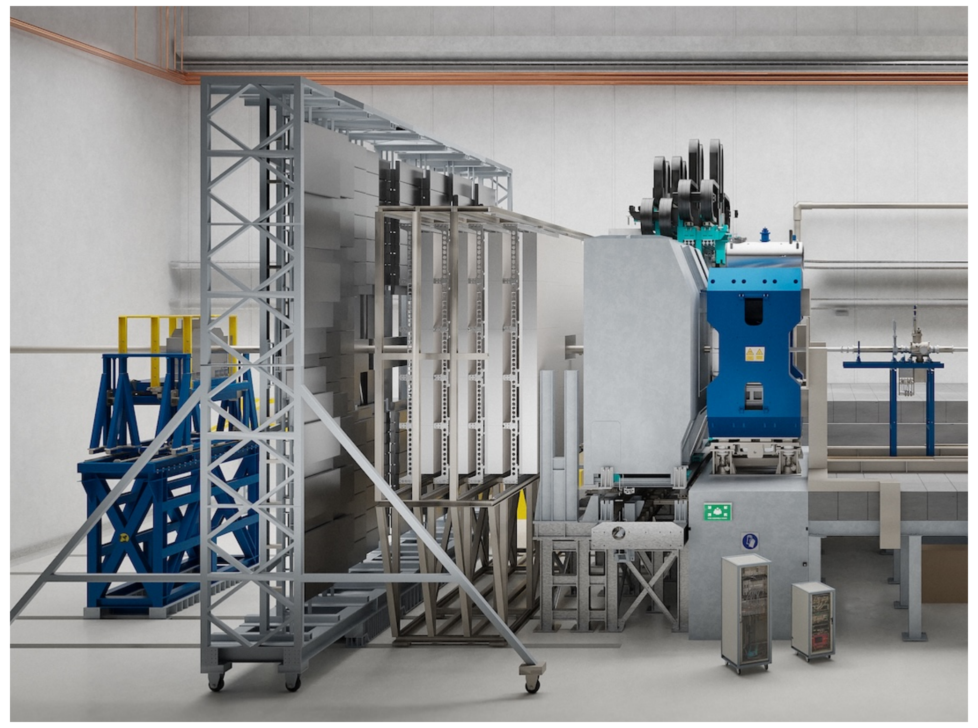
Timeslice buffer utilization

Summary: CBM online data distribution

- Key achievements
  - Timeslice/microslice data model
  - High-throughput data distribution (>1 TB/s)
  - Optimized RDMA-based zero-copy timeslice building
  - Flexible online interface using shared memory
- System validation
  - Successful full-system test at mCBM
  - Continuous use in physics and development setups
  - Peak data rates above 5 GB/s achieved, well below performance limits
  - Automated run control and monitoring implemented
- Looking forward to the start of CBM operation at SIS100

Compressed Baryonic Matter (CBM)
experiment at FAIR

- High event rates (10^7 Hz), complex (topological) trigger signatures
- Self-triggered detector front-ends, data push readout architecture



SPONSORED BY THE
Federal Ministry of Education and Research

CBM

Jan de Cuveland
cuveland@compeng.uni-frankfurt.de

~ FIN ~