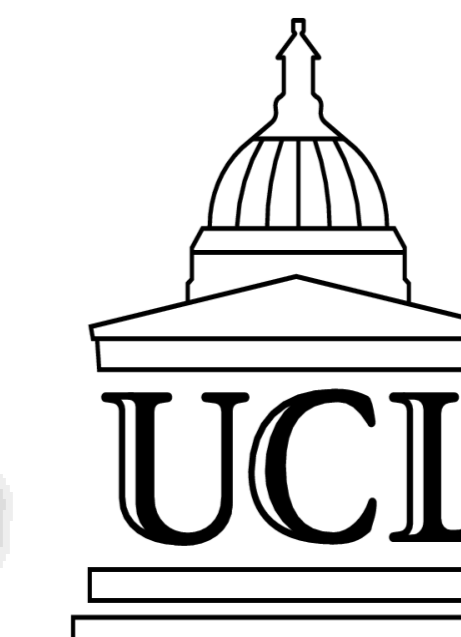


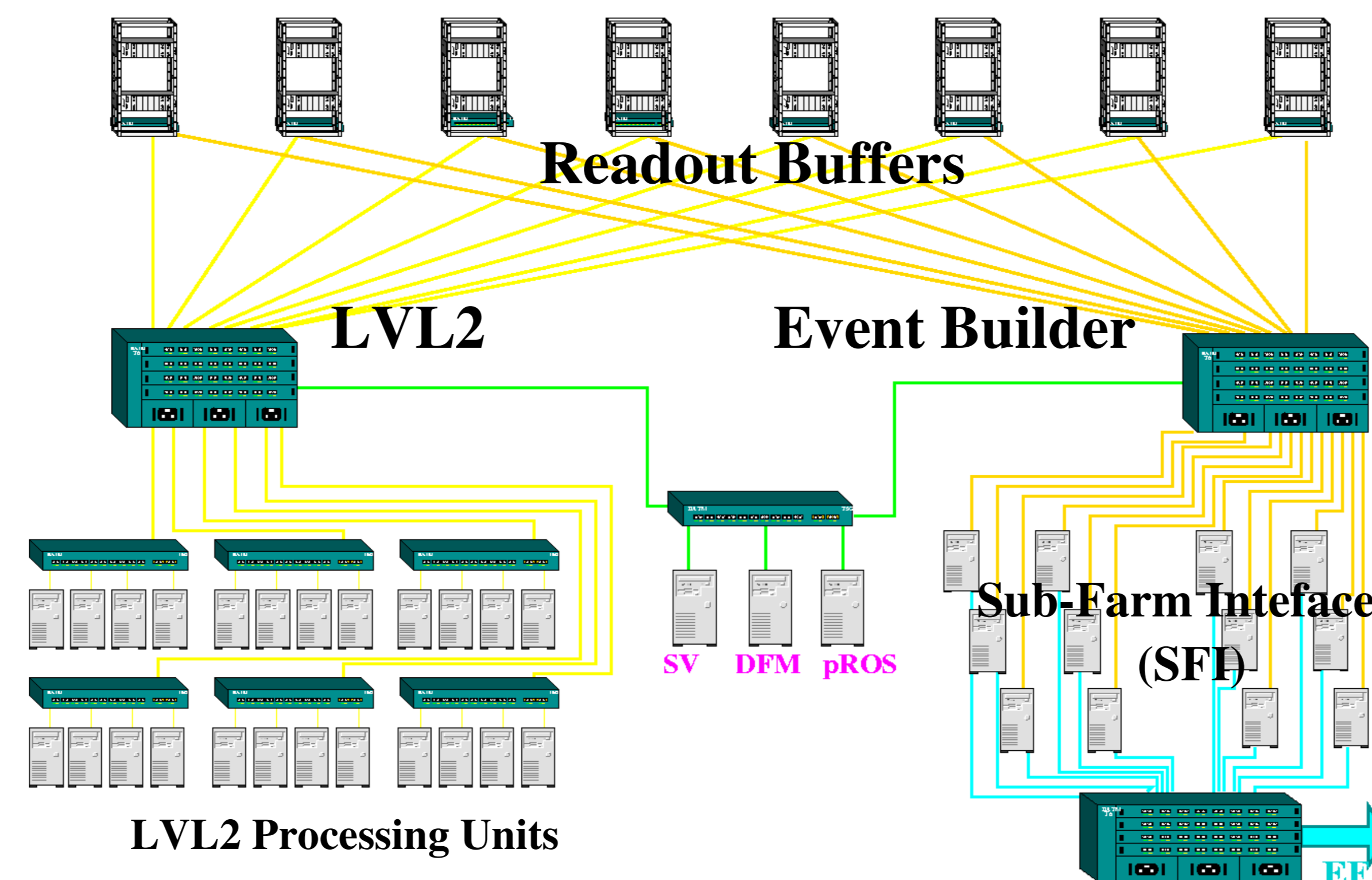
# Computer modeling the ATLAS Trigger/DAQ system performance

Robert Cranfield (UCL), Piotr Golonka (CERN/AGH), Anna Kaczmarek (IFJ), Krzysztof Korcyl (IFJ), Jos Vermeulen (NIKHEF), Sarah Wheeler (UofA)



The Trigger/DAQ system for the ATLAS detector at the CERN LHC will be based on a large local area network (LAN). More than 1600 Readout Buffers (ROBs) transmit on request detector data via the network to tens to hundreds of Level 2 Processing Units (L2PU) and approximately 150 Event Filter Subfarm Interfaces (SFI). The SFIs build complete events and pass them on request to one of several thousands event filter (EF) farm processors. Commodity Ethernet has been chosen for the network because of its cheapness, widespread usage and availability over a range of transmission speeds and physical media. The L2PUs and SFIs will be implemented as applications running on commodity PCs under the Linux OS. On the basis of information provided by the Level 1 Trigger part of the detector data from events accepted by the Level 1 Trigger and stored in the ROBs are fetched from the buffers by the L2PUs. These will run sequences of trigger algorithms and will handle a maximum event rate of 75 kHz. For the events accepted by the trigger algorithms all event data will be fetched from the buffers by SFIs to form a complete event (event building) with a rate of not more than a few kHz. The complete events are sent on request to the Event Filter farm, where off-line algorithms will reduce the trigger rate further (by approximately an order of magnitude). The expected total rate of data flowing from the ROBs to the L2PUs and EFs is at maximum about 5 GB/s.

The type of simulation implemented is known as "discrete event simulation". The simulation program maintains a time-ordered list of moments when the modeled system (or part of it) is allowed to change. We built two models of the system: one based on the Ptolemy platform, the other being a dedicated C++ program



- LVL2: Level-2 Trigger** - first asynchronous level of filtering system
- EB: Event Builder** - assembles data fragments from detector buffers into complete event and passes it to Event Filter.
- EF: Event Filter** - last level of ATLAS filtering system
- DFM: Data-Flow Manager** controls event building
- SV: Level-2 Supervisor** controls Level-2 trigger
- pROS: pseudo-ROS** - buffers results from Level-2 trigger
- SFI: Switch-Farm Interface** builds events and provides these to Event Filter processors

The behavior of the calibrated component models is in good agreement with the behavior of the real components in small test setups. The first experimental results for event building in a larger test set-up also show an encouraging agreement with the model predictions. However, further validation is required. Models for the full-scale system already allow determination of possible problem areas and investigation of techniques for preventing build-up of queues in switches and processors. In the models build-up of queues may result in long second-level trigger decision times or event-building times; message loss due to queue overflow may in reality also occur. Models for the full system, for event building based on the calibrated component models and for the LVL2 trigger so far based on "paper model" assumptions, have been run ("paper model" assumptions are the assumptions made in the calculations of average message frequencies, bandwidth requirements and CPU capacity requirements using first level trigger rates and details of the trigger processing and mapping of the detector onto the ROBs). It has been shown that a credit-based pull scenario for event building and an assignment scheme that minimizes the number of events assigned to each second-level trigger processor effectively limits queue lengths and allows a high degree of utilization of processor capacity and of network bandwidth. It has also been found that the use of small switches for connecting small groups of second-level trigger processors to a larger central switch could prove to be problematic as undesirable build-up of queues in the central switch may occur.

## Parameterization

## Calibration

## Validation

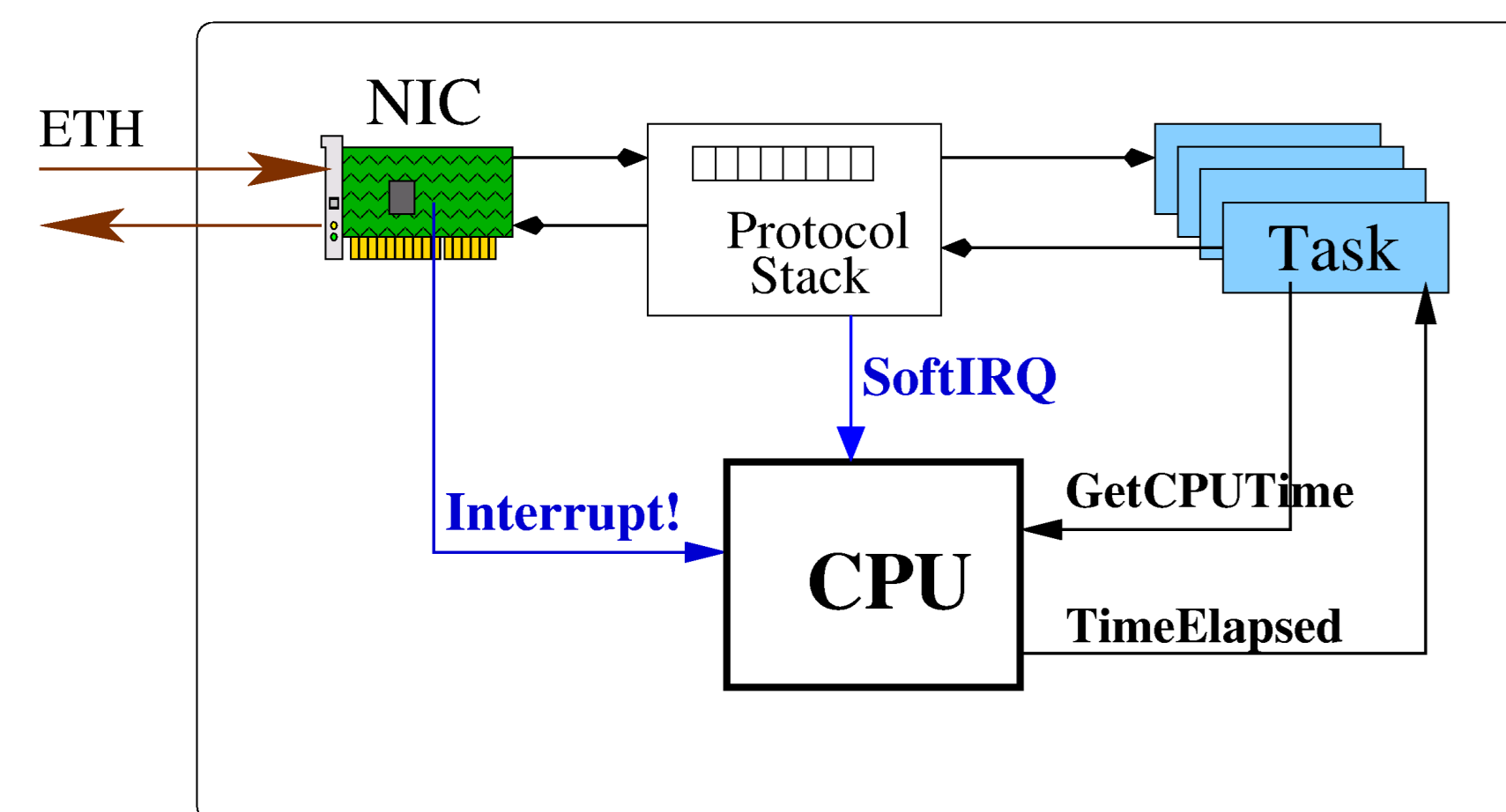
## Prediction

The models of the system components are kept as simple as possible but are sufficiently detailed to reproduce behavioural aspects relevant to the issues studied. Each model has measurable parameters.

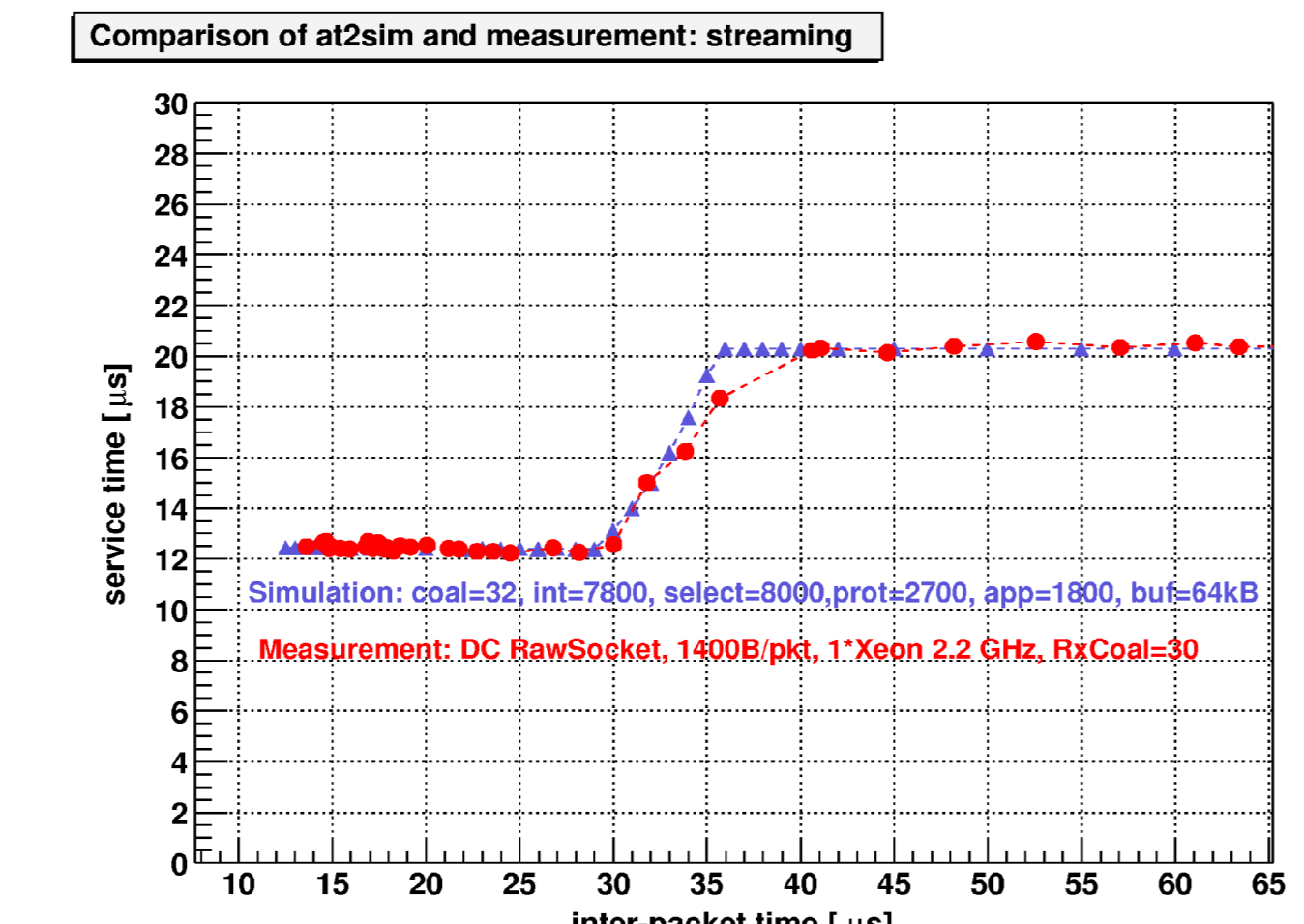
The values of the model parameters for the software processes were based on time stamps obtained from test measurements. The parameterized model of the switches was calibrated using results obtained from dedicated setups with hardware traffic generators.

Measurements results from several test setups (up to 10% of the full size system) were used to validate the parameterization and calibration.

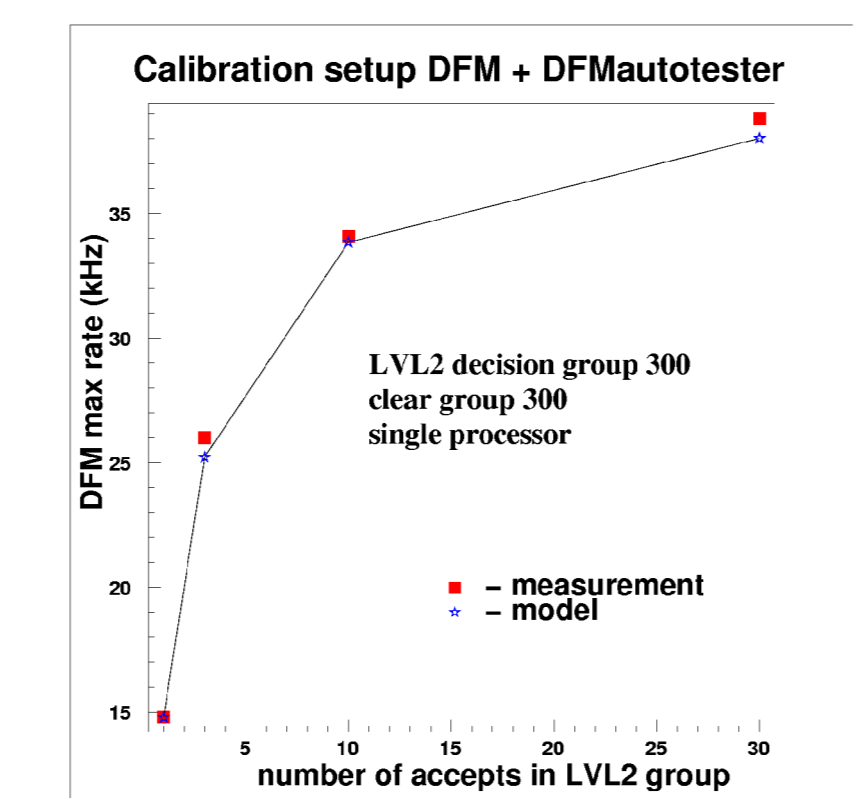
Results for the full size system were produced. Various ideas on traffic shaping aimed at improvement of performance and at avoidance of performance degradation due to packet losses were evaluated.



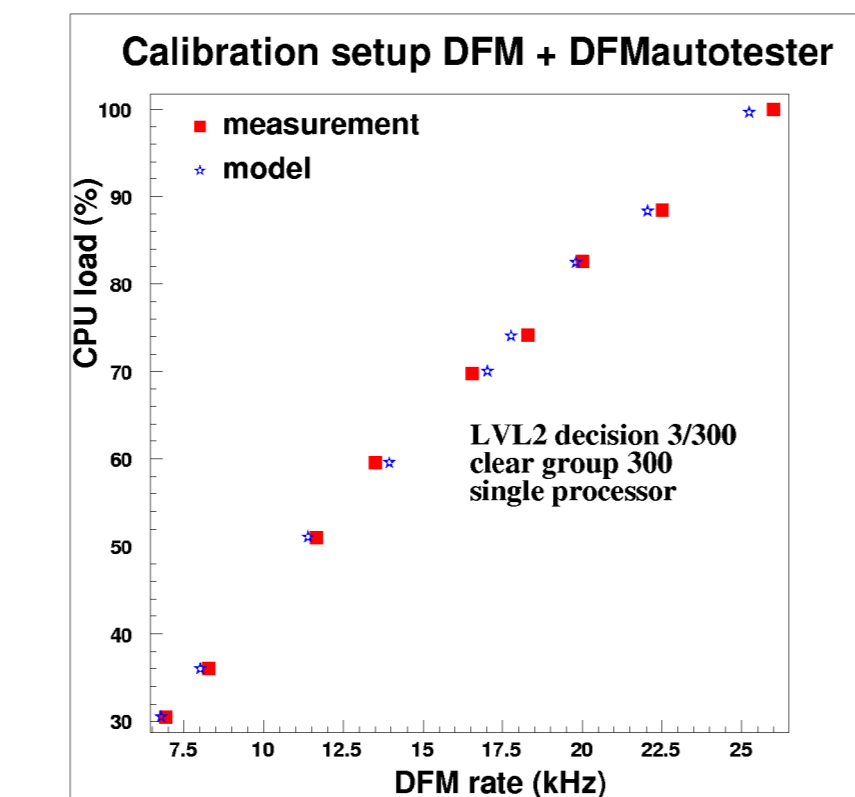
Model of a multi-tasking operating system with interrupt-driven network communication and protocol stack sharing processing time of a single processor. DFM parameterization highlighting times necessary to reproduce behaviour related to reception of the L2SV\_Decision message and generation of the DFM\_Assign Messages for the SFIs



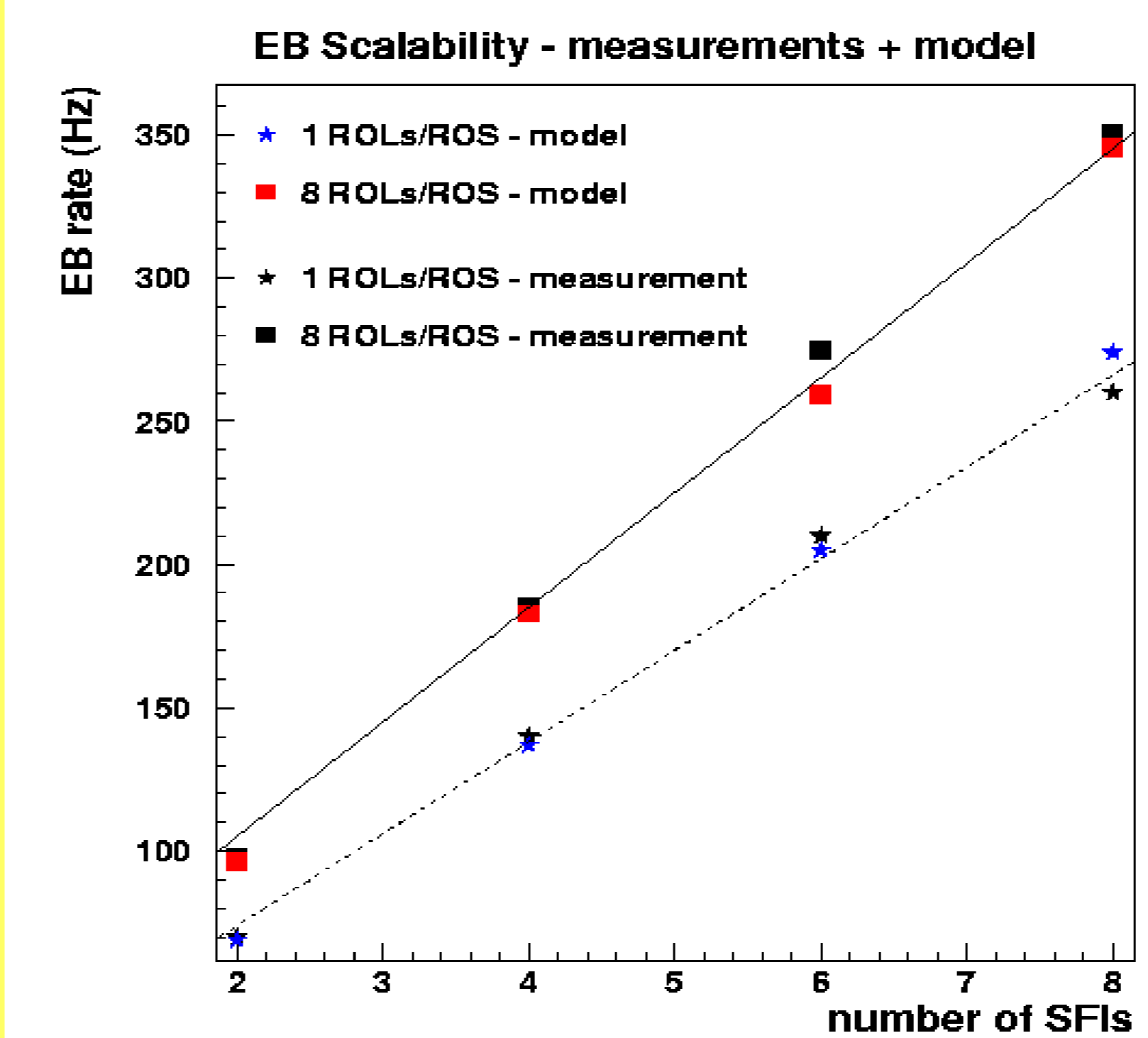
Calibration of DC low-level communication  
Interrupt coalescence (coal): 32 µs  
Interrupt service time (int): 7.8 µs  
Linux read low-level (select): 8 µs  
Linux protocol stack (prot): 2.7 µs  
Application activation: 1.8 µs



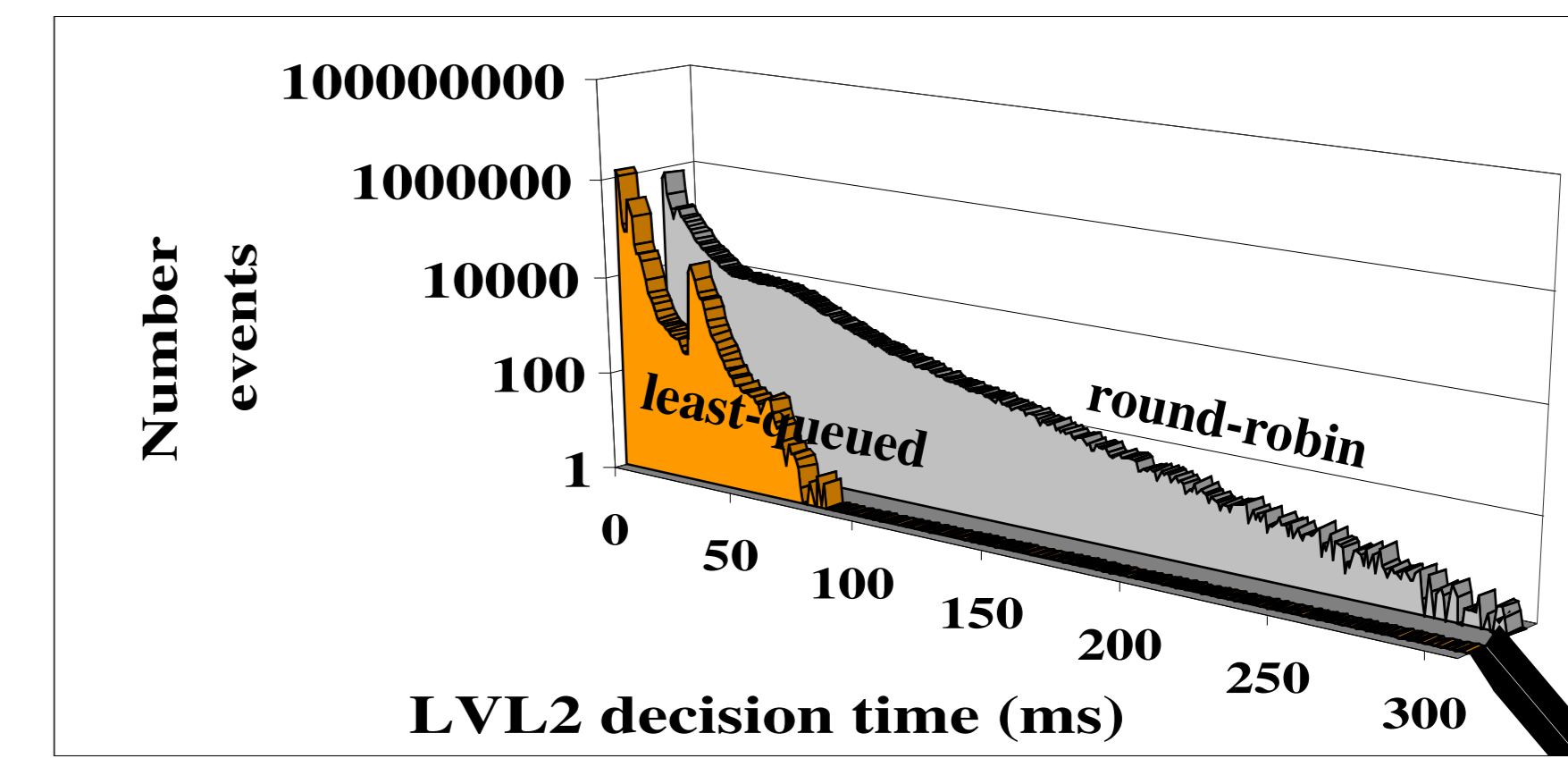
The DFM receives from the SV messages with lists of decisions produced by the LVL2 trigger: accepted events need further processing and an SFI has to be assigned, rejected events result in clear messages broadcasted to the ROBs. The maximum DFM rate depends on the number of accepted events per message from the SV (the message always contains a list of 300 events).



Model of the CPU sharing allows to estimate CPU load when running the DC application.



The biggest challenge for modeling is to predict the scalability of the final system. Correct modeling of testbeds with various sizes increases our confidence in the models used. The plot shows that the maximum event building rate scales linearly with the number of SFIs. The intercept of the line fitted for buffers with receiving data from a single Read-Out Link (ROL) (1600 network access points) is smaller than that of the line for buffers aggregating the data from 8 ROLs (200 access points), as a smaller number of request messages is required for the latter.



Decision time of the second level trigger for a model of the full system for 3.5 million LVL1 accepts, 100 L2PUs utilized at on average 83% and for round-robin (grey) assignment and assignment to the L2PU with the smallest number of events already assigned to it ("least-queued", orange). The peaks are due to the different steps in the trigger algorithms

Each SFI can limit number of active requests (credits) sent to detector buffers to avoid overflows in buffers in the switches. We predict proper credit values for smooth operation of the full system

