links, the transfer time of the subevents will be less, thus even higher event rate can be reached. On the other hand, 80-100 Mbyte/s throughput is close to the maximum value that can be achieved nowadays on the most powerful high speed links.

Simulation of an optimized and refined data acquisition system with different event building methods is the task of the near future. Simulation of different technologies on different data transfer protocol levels should also be accomplished in order to see whether the required parameter values are feasible or not.

6. References

Figure 14 shows the required Standard Detector link bandwidth values which are necessary for the 50 Hz event rate at different subevent size deviations in the [0.65 Mbyte ... 1.95 Mbyte] subevent size interval. The link bandwidth values were measured by simulation. Due to the low default Subevent size deviation, the Subevent range factor did not influence the required Standard Detector link bandwidth significantly. However, by using higher deviation, the Subevent range factor will probably have stronger effect to the bandwidth of the Standard Detector links.

5. Summary

The behaviour of the simulated data acquisition system is essentially determined by the input and output event rate of the LDCs. If the input event rate is less than the output event rate, the occupancy of each LDC buffer has a finite high bound and we can estimate the optimal size of the buffers. The optimal size depends on the mean value and the deviation of the subevent sizes, the number of LDCs and the applied event building method. For a 32x32 system, the optimal buffer size is about 50 Mbyte if we suppose 41.6 Mbyte events in average, with a deviation of a few per cent of the mean value.

If the input event rate of the LDC is greater than the output event rate, the occupancy of all the LDC buffers is increasing until one of the buffers becomes full. Afterwards the LDCs can receive less than 50 events per second depending on the throughput of the High Speed links, i.e. events will be lost in the LDCs.

The input event rate of the LDCs is limited by the link bandwidth of the Standard Detector links. The bandwidth utilisation of the Standard Detector links is determined by the maximum size of the subevents and the features of the Trigger System. Using the default values of the basic parameters resulted in 72% bandwidth utilisation of the Standard Detector links, when the High Speed links could transfer the events at 50 Hz event rate. In case of 41.6 Mbyte average event size, 300 Hz trigger rate and 1 ms communication latency in the Data Flow control link adaptors, the required Standard Detector link bandwidth is about 90 Mbyte/s. However, if the event size is greater than 45-50 Mbyte or the subevent size deviation exceeds the 10% of the mean value, the Standard Detector links require more than 100 Mbyte/s link bandwidth, i.e. the Standard Detector links seem to be a real bottle-neck of the system.

The output rate of the LDCs is determined by the throughput and the communication latency of the High Speed links and the event building method. Using 1 ms High Speed link communication latency, the minimum High Speed link throughput, at which there was no event loss, was 71 Mbyte/s. By keeping the output rate same as the input rate, the throughput of the High Speed links must be higher than that of the Standard Detector links. This extra throughput is due to the event building process, since large subevents block the switch and do not allow other smaller subevents to be transmitted. The extra throughput can be reduced by lower subevent size deviation and better event building methods.

As a conclusion, we can say that the default values of the input parameters resulted in a reasonable performance of the simulated data acquisition system. By increasing the number or the throughput of the High Speed links, the performance of the system may improve. If we increase the number of the High Speed links, the average size of the subevents will be smaller, thus more than 50 events can be saved in one second. However, the deviation of the subevent sizes will increase if events are composed of more subevents, and therefore the blocking effect of the large subevents will be relatively higher. By increasing the throughput of the High Speed
4.4 Standard Detector link bandwidth

In order to reach the 50 Hz event rate on the Standard Detector links, we had to set different bandwidths for the links, depending on the event size. The required Standard Detector link bandwidth values are shown in Figure 13. These values are determined by the FEC read-out time, the average waiting time for the next trigger signal after the FECs are all read out, and the communication latency in the Data Flow control link adaptors. The bandwidth values of Figure 13 were measured by simulation. At each event size, the bandwidth utilisation of the Standard Detector links was 72%.
cal to the 1.3 Mbyte mean value, and then we changed the deviation of the subevent sizes from 0 up to 60% of the mean value. The Event rate as a function of Subevent size deviation is shown in Figure 9. As we can see, above 20% deviation the event rate is less than 50 Hz, since due to the increased deviation, there are relatively more large subevents than at lower deviation, and large subevents causes the switch be blocked more often. Thus the average output rate of the LDCs decreases, and the LDC buffers become saturated.

Figure 9. Event rate as a function of Subevent size deviation

Figure 10. Event rate as a function of LDC buffer size

Figure 10 shows how the Event rate depends on the LDC buffer size. This curve has three different parts. If the LDC buffers are smaller than 40 Mbyte, the buffers will not be able to store at least 32 subevents of average size, therefore the event building will be blocked and the event rate will strongly depend on the buffer size. Between 40 and 48 Mbyte, the buffers are in the balancing range below the optimal size, thus only very large events will be lost. In this interval, the curve increases slowly up to 50 Hz. Above 48 Mbyte the event rate is constant 50 Hz, however the buffer space utilisation decreases, i.e. the buffers are redundant.

Figure 11 shows the Event rate as a function of the High Speed link throughput. The critical throughput is about 71 Mbyte/s (with 1 ms High Speed link communication latency as a default). Since the throughput of the Standard Detector links was 65 Mbyte/s (at 41.6 Mbyte default event size), the applied event building algorithm increased the required throughput of the High Speed links by an additional 6 Mbyte/s. This extra throughput is due to the large subevents which block the switch, and it depends strongly on the deviation of subevent sizes. This effect is the reason why the Event rate is less than 50 Hz above 20% deviation (see Figure 9). Under the critical throughput, the Event rate decreases proportionally, if the High Speed link throughput decreases.

Figure 12 shows how the Event rate depends on the High Speed link communication latency. The Event rate starts to decrease about 2.5 ms communication latency, but it is stable 50 Hz under that value. The critical value of the High Speed communication latency depends strongly on the High Speed link throughput, because if the throughput decreases, the communication latency must be decreased, in order to maintain the same event rate. For example, at 71 Mbyte/s High Speed link throughput, the High Speed link communication latency can be the default 1 ms at most if 50 Hz event rate is required.
LDC buffers, and the size of the fraction depends on the LDC buffer size. From this Figure we can predict that the optimal LDC buffer size will be about 50 Mbyte for a 32x32 system, if all the other parameters take their default value and we use an event building algorithm as mentioned before. In case of other event building algorithms, the optimal LDC buffer size may change.

In Figure 8, we can see the Event rate as a function of Subevent range factor. Since the deviation of the subevent sizes was very low (5%), and the 50 Hz event rate was feasible at the default event size, the width of the subevent range did not influence the event rate. That is why we made an additional simulation to see the effect of the changes of the subevent size deviation. We set the subevent range to the [0.65 Mbyte ... 1.95 Mbyte] interval, which is symmetri-
i.e. for a 32x32 (full), a 16x16 (half), a 8x8 (quarter) and a 4x4 (eighth) system. In order to produce these curves, we set the scaled parameters respectively, while the unscaled parameters were set to their default value. Thus both of Figures 3 and Figure 4 show only the scalability of the simulation and not the real performance of the simulated systems. In these Figures, we can see that simulating a 16x16 system instead of a 32x32 system means less than 1% difference at the Event rate, and less than 4% difference at the Average total event latency, so the results of the 16x16 system give a good approximation of the performance of a 32x32 system.

In Figure 3, Figure 4, and in all of the following Figures, the parameters are scaled corresponding to a 32x32 system, as well as in the explanation of the results below. It means that the event sizes, the buffer sizes and the Average total event latency values are multiplied by two, while all the other parameters are the same as during the simulation.

4.3 Event rate

Figure 5 shows the Event rate as a function of Event size at different scaling factors. Figure 6 shows how the Event rate depends on the Event size if different LDC buffers are used. At 120, 80 and 60 Mbyte LDC buffer, the event rate is constant 50 Hz under the critical event size, which is about 46 Mbyte. Above the critical event size, the event rate decreases proportionally to the Event size. At 40 Mbyte LDC buffer, the critical event size is about 36 Mbyte. In order to reduce the effect of the initial phase (i.e. while the buffers are getting full) when 120 Mbyte LDC buffer was used, we simulated 100 s data acquisition.

Figure 7 shows the Average total event latency as a function of Event size at different LDC buffers. The fraction in the curves at the critical event size is due to the saturation of the

Figure 3. Event rate as a function of Event size at different scaling factors

Figure 4. Average total event latency as a function of Event size at different scaling factors
tested during the simulations. We defined an “almost full” state for the LDC buffers to avoid the buffers to overflow. An LDC buffer was regarded as almost full, if it had less free space than the maximum size of one subevent. If an LDC buffer got into the almost full state, the following events were discarded until the buffer occupancy decreased under the almost full level. These discarded events were regarded as lost events.

The buffer size of the Global Data Consumers was 64 Mbyte that was enough for two events of maximum size, since a GDC had to be able to receive the subevents of a new event while it was transmitting the whole previous event to a Permanent Data Storage.

The data acquisition system is based on a 32x32 switch, but actually we simulated only the half of this switch. We set the internal latency of the switch to 20 µs.

The PDS links between the Global Data Collectors and the Permanent Data Storages were supposed to have the same features as the High Speed links.

The Event Destination Manager had two functions: to control the event building process and to generate trigger signals to the LDCs to read out the FECs. In order to separate the event building process from the control and trigger operation, we supposed an additional Data Flow control link between the EDM, the LDCs and the GDCs. The Data Flow control link was used only to send “free” signals from the GDCs to the EDM, and to send multicast read-out trigger signals to the LDCs. Since these messages are very short messages (typically one frame or packet), the transfer time of these messages was set to zero, but we supposed 1 ms communication latency in the Data Flow control link adaptors.

The event building algorithm applied in the simulations works as follows. If there is a free GDC, the first LDC starts to send its corresponding subevent to that GDC. During the transmission the switch port, which is connected to the destination GDC, is reserved. Each LDC can start to transmit a subevent, if the previous LDC has finished to transmit the subevent of the same event. If there is no free GDC, each LDC buffers the new subevents. After the system has become stable, this algorithm works similar to the Barrel-shifter [5].

4. Simulation results

4.1 Input and output parameters

We have generally simulated 30 seconds data acquisition which corresponded to 1500 events of average size at 50 Hz event rate. We had six input parameters: the Event size, the Subevent size deviation, the Subevent range factor, the LDC buffer size, the High Speed link throughput and the High Speed link communication latency. Only one of them was changed, while all the other parameters were set to the default value. The output parameters of the simulations were the Event rate at the output of the LDCs and the Average total event latency. The Total event latency was defined as the time from the event generation until the whole last sub-event was received by the destination GDC. The Average total event latency was calculated from the Total event latency of all built events. The Event rate at the output of the LDCs was the same as at the GDCs, since there was no event loss either in the switch or in the GDCs.

4.2 Scalability

First we checked whether the system is scalable or not, i.e. how the performance changes if we simulate only a certain part of the system. Figure 3 shows the Event rate and Figure 4 shows the Average total event latency as a function of Event size at different scaling factors,
3. Simulation setup

The simulated architecture of the ALICE DAQ system is shown in Figure 2, where dashed lines correspond to zero delay, and arrows represent the flow of events and control messages (GDC free signals, FEC read-out trigger signals for the LDCs). It is important to note, that only a 16x16 subsystem was simulated instead of the 32x32 system because it required much less CPU time and memory to run. Therefore the total event sizes and the buffer size of the Local Data Concentrators and Global Data Collectors were divided by two. However, this reduction did not influence the subevent sizes and any other input parameters. The LDC buffers must have been rescaled, since they had to be able to buffer at least so many subevents as many LDC was in the system, otherwise the event building process was blocked due to the algorithm mentioned before.

The physical events and the trigger signal of the events were generated by the Event Generator (EG). The Event Generator produced subevents of different size for each FEC. The distribution of subevent sizes followed Gaussian distribution with a standard deviation ($\sigma_e$) of 5% of the mean value. The default mean value of the subevent sizes was set to $\mu_e=1.3$ Mbyte, which corresponds to 41.6 Mbyte event size in case of a 32x32 system. Only a part of the whole event size distribution function was taken into account because the size of the real physical events are limited. So we cut a range from the Gaussian distribution curve around the mean value. The width of this range was defined by a “subevent range factor”: the low bound was the mean value divided by the subevent range factor, while the high bound was the mean value multiplied by the subevent range factor. Although the two half subintervals (above and under the mean value) were not equal, the effect of this difference was negligible due to the characteristics of the Gaussian distribution, if the standard deviation was $\sigma_e=5%$. The Subevent range factor was set to 1.1 as a default value, thus the subevent range included $2\sigma_e$ wide environment around the mean value, which contained the major part of the area under the curve. The distribution of the time interval between two consecutive trigger signals followed Exponential distribution with a mean value of $\mu_t=3.33$ ms (300 Hz).

The FECs had buffers only for one subevent. Until all of the FECs were not read out by the LDCs, the trigger signals were discarded, so the input event rate of the Local Data Concentrators was mostly determined by the throughput of the High Speed links. The bandwidth of the Standard Detector links between the FECs and the Local Data Concentrators was always set to a value to be able to transfer 50 events of average size per second, thus the maximum available event rate at the input of the LDCs was always 50 Hz, which is the required event rate for statistics at the ALICE experiment.

Between the LDCs and the GDCs, we supposed High Speed links that can be implemented by several technologies like Fibre Channel, ATM, SCI, etc. The default value of the High Speed link throughput was set to 80 Mbyte/s, since this value was a little bit higher than the minimum throughput, which was necessary to reach the 50 Hz event rate on the High Speed links while using all the other basic parameters. (The bandwidth utilisation of the High Speed links depends on the applied technology.)

The Local Data Concentrators, the Global Data Collectors and the Permanent Data Storages were supposed to be workstations with 1 ms communication latency as a default value. The communication latency is defined as the time from when an application (on the source side) sends a message with 1 byte application data till when another application (on the destination side) receives that message [4].

The Local Data Concentrators had 40 Mbyte buffer in default, it was enough for more than 16 subevents of average size, even in the case of the largest subevent size (1.8 Mbyte)
2. General architecture of the ALICE DAQ system

The general architecture proposed for the ALICE DAQ system [3] is shown in Figure 1. The system consists of Front-End Crates (FEC), Local Data Concentrators (LDC), Global Data Collectors (GDC), Permanent Data Storages (PDS), a Switch and an Event Destination Manager (EDM). The Detector and the Trigger System is not explicitly included in the data acquisition system.

The Front-End Crates contain the front-end electronics (ADC, TDC, etc.) required for the read-out of each subdetector.

![Figure 1. General architecture of the ALICE DAQ system](image1)

![Figure 2. Simulated architecture of the ALICE DAQ system](image2)

The Local Data Concentrators constitute the interface between the subdetector front-end electronics and the central DAQ system. The LDCs receive Level 1 trigger signal and forward formatted data to the Global Data Collectors through the Switch. From the LDCs to the Switch and from the Switch to the GDCs, High Speed links will be used that are available for high performance data transmission. Several technologies can be implemented, such as Fibre Channel, ATM or SCI.

The Switch performs routing of the subevents to the corresponding GDC.

The Global Data Collectors merge together the subevents of different subdetectors into one global event, and then they record the events in Permanent Data Storage devices.

The Event Destination Manager is responsible for the event building scheme used in the data acquisition system. In case of a new trigger signal, the EDM assigns the event to one GDC which will be used to build the corresponding event.

The event building scheme, which is defined both by the configuration of the system and by the applied event building algorithm, plays also a very important role in the data acquisition. One of the final aims of all simulations is to choose the most efficient event building algorithm with an optimal system configuration.
Simulation of the ALICE DAQ System
(version 1.0, September 30, 1995)

Gabor Harangozo¹, Pierre Vande Vyvre¹, Erik van der Bij¹
Gyorgy Rubin², Bin Wu³, Bjarne Kvamme³

¹CERN/ECP, Geneva, Switzerland
²KFKI-RMKI, Budapest, Hungary
³University of Oslo, Norway

1. Introduction

The Heavy Ion Experiment ALICE is one of the three future experiments in the LHC at CERN. Modelling and simulation of such a large data acquisition system is necessary in order to study the behaviour of the system.

An object-oriented program (ALSIM) [1] has been developed to model and simulate a generic ALICE DAQ system. Due to the object-oriented approach, the generic model can be extended to technology dependent implementations, such as Fibre Channel, ATM, SCI, etc. The architecture and the input parameters of the system are read from a text format configuration file, thus one can easily change the simulation setup.

The simulator program ALSIM (ALICE Data Acquisition System Simulator) is written in MODSIM II programming language [2], which is a general purpose, modular, high-level programming language for object-oriented programming and discrete-event simulation. ALSIM provides a graphical user interface, which includes histograms of some output parameters, such as Event number distribution over Total event latency time, Buffer Occupancy over simulation time, etc.

The aim of the simulations was to see how the system performance depends on different values of some basic input parameters. During the simulations one parameter was generally changed, while the others were set to their default value. It means that we took a basis point in a multidimensional parameter space and examined the system performance while moving along one dimension in the environment of the basis point. The results of these simulations have given a global view of the parameter space around the basis point, which makes it easier to find an optimal value set of the input parameters later on.

In this paper, first we introduce the general architecture of the ALICE DAQ system, and we describe the simulation setup with the default values of the input parameters. Then we present the simulation results, and finally we summarize the results of the performance simulations.