THE PHENIX EVENT BUILDER

David Winter[*], Columbia University, New York, NY 10027, USA
for the PHENIX Collaboration[†]

Abstract

The PHENIX detector consists of 14 detector subsystems. It is designed such that individual subsystems can be read out either independently in parallel or as a single unit. The DAQ used to read the detector is a highly-pipelined parallel system. Because PHENIX is interested in rare physics events, the DAQ is required to have a fast trigger, deep buffering, and very high bandwidth. The PHENIX Event Builder is a critical part of the back-end of the PHENIX DAQ. It is responsible for assembling event fragments from each subsystem into complete events ready for archiving. It allows subsystems to be read out either in parallel or simultaneously and supports a high rate of archiving. In addition, it implements an environment where Level-2 trigger algorithms may be optionally executed, providing the ability to tag and/or filter rare physics events. The Event Builder is a set of three Windows NT/2000 multithreaded executables that run on a farm of over 100 dual-cpu 1U servers. All control and data messaging is transported over a Foundry Layer2/3 Gigabit switch. Capable of recording a wide range of event sizes from central Au-Au to p-p interactions, data archiving rates of over 400 MB/s at 2 kHz event rates have been achieved in the recent Run-4 at RHIC. Further improvements in performance are expected from migrating to Linux for Run-5. The PHENIX Event Builder design and implementation, as well as performance and plans for future development, will be discussed.

INTRODUCTION

Heavy ion collisions provide an ideal system for studying the behavior of strongly interacting matter under extreme conditions. The prospect of observing a new state of matter, such as the quark-gluon plasma (QGP) has made the study of heavy-ion collisions one of the most exciting fields in physics today[1]. The state of the art of such collisions is represented by the Relativistic Heavy Ion Collider (RHIC) at Brookhaven National Laboratory.

RHIC has two independent rings of 3.83 km in circumference, capable of colliding nearly any nuclear species against nearly any other nuclear species. It is capable of delivering center-of-mass energies of up to 200 GeV/c per nucleon for heavy species (eg. Au-Au) and 500 GeV/c for p-p collisions. The interaction rate at RHIC is very high; the beam crossings take place at 10 MHz, while collisions take place at approximately 10 kHz (for Au-Au).

The PHENIX spectrometer resides in one of the five experimental halls in the collider. It consists of 14 detector systems in four arms. At the center are detectors for event characterization. Two central arms designed for detecting hadrons, photons, and electrons sit at mid-rapidity. At forward rapidities are two arms for tracking and measuring muons.

The high rates of high-energy collisions provided by RHIC, combined with the diversity of detectors in the spectrometer, place strong design constraints on the PHENIX DAQ. Thus it is a parallel pipelined design, containing deep buffering, high-bandwidth, and fast processing.

An overview of the PHENIX detector can be found in [2]. The basic unit of organization of the PHENIX DAQ is the granule (see Figure 1), which can be a detector subsystem or part of a detector subsystem. One or more granules read out as a unit constitute a partition. Some granules can be used to make fast Level-1 decisions, which in turn are used to signal the front-end electronics to digitize the data in the detector. This data is passed up to the Data Collection Modules (DCMs) and their supporting hardware, where early processing (for example, zero-suppression) takes place. At this stage, the data enters the Event Builder.

THE EVENT BUILDER

The PHENIX Event Builder is a set of three programs run on a cluster of x86 servers: the SubEvent Buffer (SEB), the Event Builder Controller (EBC), and the Assembly and Trigger Processor (ATP). The SEB's primary job is to collect the data from a single granule, termed a ”subevent”. The EBC receives event notification and assigns the event to an available ATP for assembly. The ATP is responsible for making data requests to the SEBs for the events assigned to it, assembling them, and writing them to short-term storage. Optionally, the ATP is responsible for making Level-2 decisions since the entire event is available to it. Once the EBC has been notified of event completion, it tells the system to flush the data from its buffers.

Hardware

The Event Builder computing cluster consists of 105 rack-mounted 1U x86 servers. There are three generations of servers from two vendors: 28 Dell[4] PowerEdge 1000s, and 77 servers from Microway [3] in two generations. The PowerEdge servers have dual 1.0 GHz Pentium III CPUs, 256 MB ECC RAM, and 8 GB Seagate U160 SCSI hard

[*] winter@nevis.columbia.edu

[†] For full author list and acknowledgements, see Ref.[1]

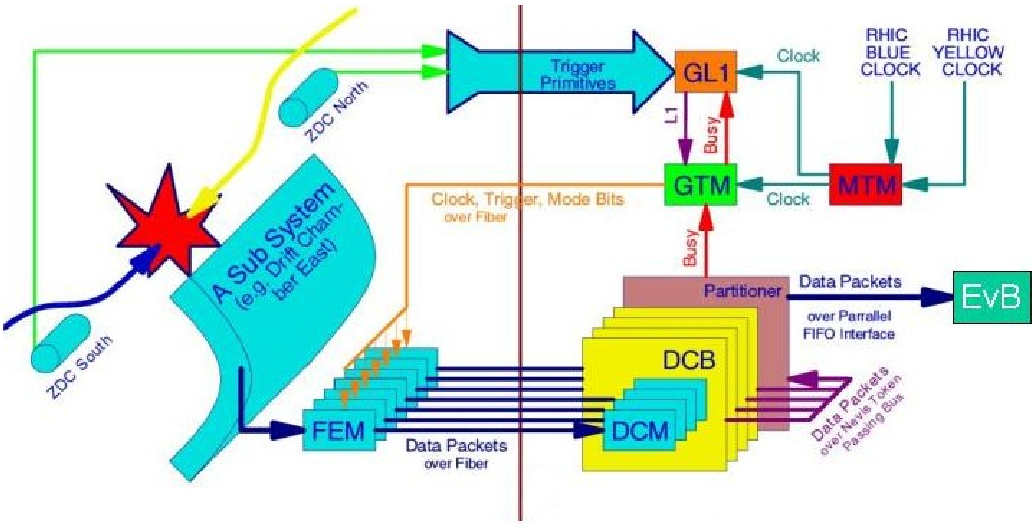

Figure 1: Schematic of a granule in the PHENIX DAQ.

drives. The newer Microway servers have dual 2.4 GHz Pentium 4 Xeon CPUs, 1.0 GB ECC RAM, and 37 GB Seagate ST340016A ATA100 hard drives. Depending on the motherboard used, Tyan or Supermicro, these machines have 400 or 533 MHz FSBs, respectively.

All the servers have dual on-board ethernet controllers. The PowerEdge servers were shipped with dual Intel Fast Ethernet interfaces, though 17 have been retrofitted with Gigabit adapter cards (16 Intel MT 1000 Server Adapters and one Syskonnect SK-98xx). The older Microway machines have combination Fast Ethernet/Gigabit dual interfaces, while the newer ones shipped with dual Gigabit interfaces. All Gigabit controllers in the Microway machines are the Intel MT 1000 Server Adapters.

The switch servicing the cluster is a Foundry FastIron 1500 Gigabit switch, providing a switching capacity of up to 480 Gbps. It has 15 modular slots, 10 of which have been used. We have two control modules providing 24 Fast Ethernet ports each and six 16-port Gigabit boards. All 96 Gigabit ports were used in Run-4. In addition, we enabled jumbo frames on the switch, using an MTU of 9014.

The core of the SEB is built around the interface between it and the data stream from the granule. This interface is implemented as a PCI adapter called the JSEB card. The JSEB is a 32-bit PCI card capable of either 3.3v or 5v signalling. It has an Altera Flex10k FPGA. It's input is a 50-pin 32-bit parallel cable. One of its important features is dual banks of 1 MB RAM, thereby allowing the card to write data to one bank while the CPU reads from the other. The PCI driver used is WinDriver from Jungo [5], while the firmware for the FPGA was written by PHENIX collaborators. The card is capable of DMA burst mode, however this mode will first be available for the upcoming Run-5. Figure 2 shows the transfer characteristic for JSEB read operations.

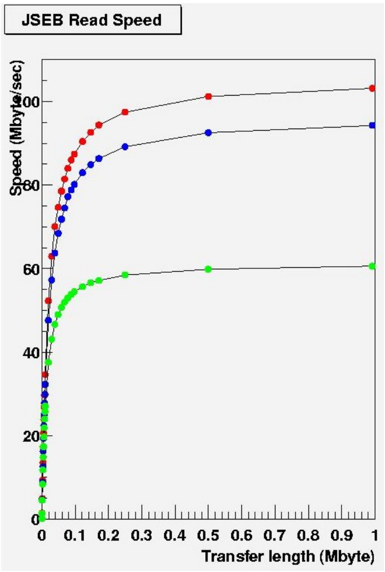

Figure 2: JSEB read speed (in MB/sec) versus read size. The green circles (lowest curve) are data taken with no DMA mode enabled. The blue circles (middle curve) are with 4-word DMA burst mode, and the read circles (top curve) are with continuous DMA burst mode. Plot is courtesy of John Haggerty.

Software

Currently the software environment used for development and deployment of the Event Builder is undergoing a shift. Up to and including Run-4, the Event Builder ran under Windows NT/2000, using Visual C++ 6.0. Control communication between components is accomplished using Iona's implementation of CORBA (ASP 6.0). Java is used for user interfaces, and the Boost template library for non-trivial algorithms and objects. The major change for Run-5 was the decision to move to the Linux platform. The distribution chosen is FNAL's Scientific Linux, which includes g++ 3.2.3. Among other advantages, Linux has enabled a distributed development environment, including the use of distcc to reduce the compilation time from 30 minutes to less than three (for 90 thousand lines of code over 170 compilation units).

Design

The three Event Builder components (the SEBs, ATPs, and EBC) are derived from a single basic design. The super class is a state machine with four states: booted, configured, initialized, and running. CORBA is used to initiate state transitions for, pass configuration data into, and fetch monitor data out of, each component.

Both control and data communication between components takes the form of messages, and under the Win32 version all network I/O was done using UDP sockets. This approach was chosen for simplicity and speed, as well as being originally driven by the choice of ATM as the underlying network technology. Under Linux, however, we have moved the control messaging to TCP. This has, in fact, simplified much of the code, as we no longer need timeout mechanisms to compensate for the unreliability of the UDP transport. Broadcast messages (eg. flush messages) are sent via multicast in both the Win32 and Linux versions of the Event Builder.

Performance and Monitoring

An important feature of the Event Builder components is that they internally keep statistics that describe their state. Examples include such parameters as number of events processed, I/O bandwidth, and error counters. CORBA methods have been defined that allow user code to query a component for the current values of its monitor data. A profiler has been written in Java that displays these parameters. It updates theses parameters at a rate of 1 Hz. In addition to a simple panels displaying these values, stripcharts as a function of time, as well as histograms as a function of component, are available for each parameter.

These tools allowed a critical examination of the performance of the Event Builder in Run-4. At peak performance, the Event Builder was capable of assembling events over 200 kB in size, logging data at up to 400 MB/s, and achieving data rates as high as 2.2 kHz. Typical running conditions were in the 170-180 kB, archiving 320-380 MB/s at a rate of 2 kHz. This performance set PHENIX data-taking records (in fact, the collection of 1.5 billion events in Run-4 would have previously been impossible), but comparing it to the collision rate RHIC provides (10 kHz), it is woefully below target.

FUTURE DEVELOPMENT

Late in the Spring of 2004, it was concluded that the most obvious improvement we could make between Run-4 and Run-5 is to move the Event Builder to the Linux platform. The original choice of Win32 was driven by the need to use ATM, the state of the art in networking when this project was started. Since the beginning of Run-4, all ATM code was removed in favor of Gigabit Ethernet. We were then in a position to switch from a special purpose platform (from the PHENIX perspective) to one that shares a common development environment with the rest of the experiment. This switch also represented an opportunity to re-evaluate the architecture.

The basic design was considered sound, so a direct porting strategy was adopted. The toughest issues that arose were regarding asynchronous I/O and Win32's Events. The latter was successfully replaced with a framework based on Boost condition variables, the details of which are beyond the scope of this discussion. Win32's overlapped socket I/O was critical to achieving the performance of Run-4. Unfortunately, there is little to no support for asynchronous socket I/O in the normal Linux kernel. The lack of asynchronous I/O eventually proved to be a feature, as it simplified the network code significantly. The important question is still what impact this I/O has on the performance.

Figure 3 quantifies this impact. This graph plots the write bandwidth versus the write size using a UDP socket. It compares the performance of Linux sockets with the three possible mechanisms available under Win32: synchronous, asynchronous, and asynchronous with callback. The implication is striking: Linux network I/O doesn't simple outperform Win32, it does so drastically. This result is encouraging, since one concern was that synchronous I/O in Linux would incur too much overhead.

To date, all three components have been ported to Linux and are running stably. All I/O has been implemented synchronously. Data streams for small messages (such as control) have been converted to use TCP instead of UDP. Both of these changes have simplified the framework, thereby making the coding less error-prone. In addition, 35% of the Event Builder nodes have been converted to Linux. This has had strong positive impact on the development by vastly reducing the development cycle via the use of GNU's distcc.

Preliminary tests have been performed using one- and two-granule configurations. Using simulated data as input to the Event Builder system, single-granule tests with the Level-1 granule (ie. the SEB that records all the global trigger data) have run stably as high as 26 kHz. Similar tests with two granules (Level-1 plus one detector) have run at 17 kHz. While the input to the system is simulated and tests on larger systems are waiting to be completed, it is clear that the goal of being able to archive whatever data the DAQ presents is well within our reach.

SUMMARY

Heavy-ion collisions at the Relativistic Heavy Ion Collider are an ideal system to study hadronic matter at extreme densities and temperatures. The high interaction rate

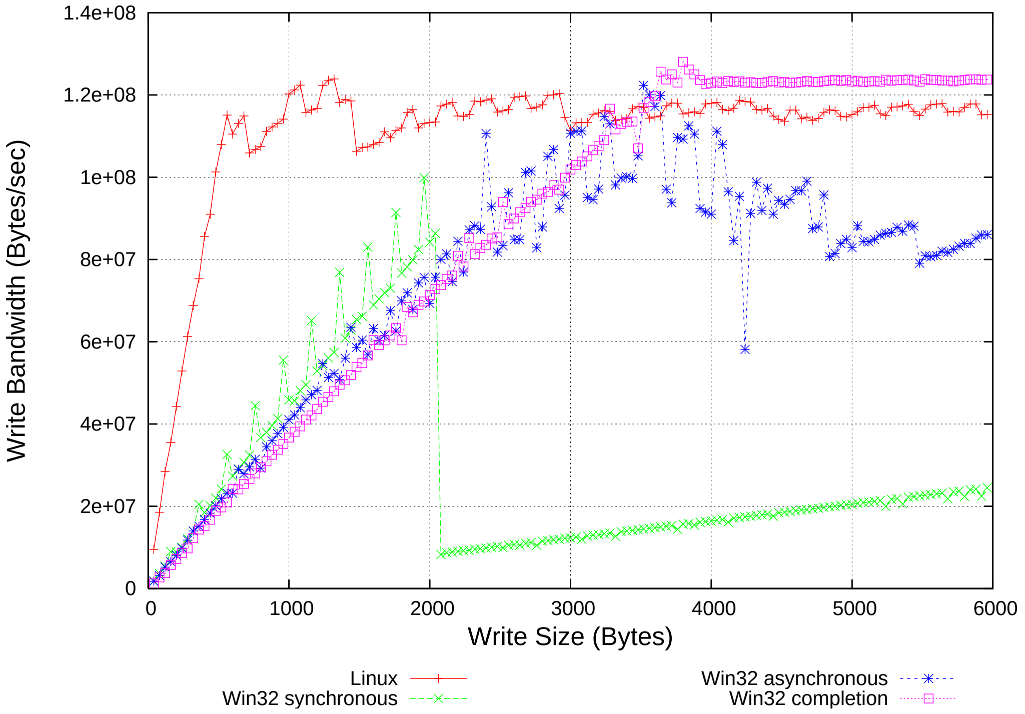

Figure 3: Comparison of write bandwidth versus write size for UDP sockets. The red crosses are Linux, while the green x's, blue asterisks, and magenta squares are Win32 synchronous, asynchronous, and asynchronous with callback, respectively

at RHIC combined with large-multiplicity events and the PHENIX need for high statistics presents a difficult challenge in DAQ and Event Builder design. The PHENIX Event Builder addresses these challenges by distributing the tasks across a cluster of over 100 servers connected via a high-capacity switched Gigabit network. Previous versions of the Event Builder have been developed under Windows NT/2000 where archiving rates of up to 400 MB/s and 2.2 kHz have been achieved. To address the increasing need to archive at even higher rates, the Event Builder has been ported to Linux and is undergoing initial validation testing. Initial tests have shown event rates as high as 26 kHz are achievable. Run-5 at RHIC will have a Linux-based Event Builder that will meet or exceed the desired archiving rates.

REFERENCES

[1] K. Adcox et al., “Formation of dense partonic matter in relativistic nucleus-nucleus collisions at RHIC: Experimental evaluation by the PHENIX collaboration”, nucl-ex/0410003

[2] K. Adkox et al., “PHENIX Detector Overview”, NIM A499 469-479 (2003)

[3] http://www.microway.com

[4] http://www.dell.com

[5] http://www.jungo.com