Journal logo

Development of a Data Acquisition Sub-System using DAQ-Middleware

K. Nakayoshi[a*], Y. Yasu[a], E. Inoue[a], H. Sendai[a], M. Tanaka[a], S. Satoh[a], S. Muto[a], N. Kaneko[a], T. Otomo[a], T. Nakatani[b], T. Uchida[c]

[a]*High Energy Accelerator Research Organization, 1-1 Oho, Tsukuba, Ibaraki, 305.0801, Japan*

[b]*J-PARC Center, Japan Atomic Energy Agancy, 2-4 Shirane Shirakata, Tokai, Naka, Ibaraki 319.1195, Japan*

[c]*The University of Tokyo, Hongo, Bunkyo-ku, Tokyo, 113.0033, Japan*

Elsevier use only: Received date here; revised date here; accepted date here

Abstract

A prototype of a data acquisition (DAQ) sub-system has been developed at the Material and Life Science Facility (MLF), the Japan Proton Accelerator Research Complex (J-PARC). This prototype was built using software components called DAQ-Components in DAQ-Middleware. The DAQ sub-system attained scalability and flexibility, resulting from the nature of the DAQ-Components, their location transparency. The DAQ sub-system can be easily extended to keep in step with the increase in the beam intensity of J-PARC. Results of beam tests using the DAQ-Components in September 2007 certified the required fundamental DAQ functions of DAQ-Middleware.

Keywords: Data acquisition system; Network based DAQ system, DAQ-Middleware

PACS: 62.50.+p

1. Introduction

Until 2006, the High Energy Accelerator Research Organization (KEK) Neutron Science Laboratory (KENS) had performed neutron scattering experiments using pulsed neutron beams. This involved the creation of histograms in front-end electronics for data reduction [1].

At the Material and Life Science Facility (MLF) in the Japan Proton Accelerator Research Complex

Corresponding author. Tel.: + 81-29-864-5200(ext. 4765) ; fax: + 81-29-864-3284 ; e-mail: kazuo.nakayoshi@kek.jp

(J-PARC) under normal conditions, however, all data will be read from detectors without data reduction. Adoption of this method results in advanced and flexible data analysis, but the amount of data is too large to handle easily. The read-out hardware scheme being used at KENS will become a bottle-neck because of the serial read-out using VMEbus.

To solve this problem, we adopted Ethernet (100BASE-TX) based SiTCP [2, 3] in the read-out modules, called NEUNET [4], because of the parallel read-out capability of Ethernet. The new read-out scheme required a network-based data acquisition (DAQ) software framework.

Therefore MLF decided to adopt DAQ-Middleware for their DAQ sub-system with NEUNET modules.

2. DAQ-Middleware

We previously described the development of DAQ-Middleware [5, 6] extended from RT-Middleware [7].

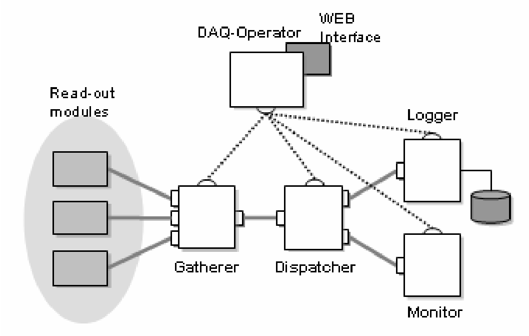

Fig. 1. Fundamental DAQ-Components of DAQ-Middleware. This figure shows the fundamental DAQ-Components and one of their configurations. Solid lines show data paths and dotted lines show command paths.

DAQ-Middleware is a framework for network based DAQ software and it is build from DAQ-Components [5, 6]. Fig. 1 shows the fundamental DAQ-Components of DAQ-Middleware and one of their configurations. They have minimum and essential DAQ functions and consist of a Gatherer, a Dispatcher, a Logger, a Monitor and a DAQ-Operator. The solid lines show data paths and the dotted lines show command/status paths. The Gatherer is an interface between the read-out hardware and the DAQ-Middleware. That is, it reads data from read-out hardware, converts it to a DAQ-Middleware format and sends it to the Dispatcher. The Dispatcher in turn receives data from the Gatherer and sends it to the Logger and the Monitor. The Logger receives data from the Dispatcher and stores it to a hard-disk, whereas the Monitor, which also receives data from the Dispatcher, analyzes the data, creates histograms and displays them. The DAQ-Operator controls other DAQ-Components. It parses a configuration file that describes configuration parameters of each DAQ-Component. In addition, the DAQ-Operator sends parameters to each DAQ-Component at the "configuration state" [5, 6]. It also includes an interface to communicate with a WEB server.

Users are no longer required to perform DAQ programming, such as network and IPC (Inter Process Communication) programming by introducing DAQ-Middleware. Rather, users can concentrate on specific programming, such as online data analysis programming.

We describe here the features of DAQ-Components, the System Configuration, and the System interface.

2. 1 Features of DAQ-Components

The DAQ-Components have data transfer, run control and system configuration functions. Users can create a new DAQ-Component to implement the core-logics of a DAQ-Component skeleton (Fig. 2).

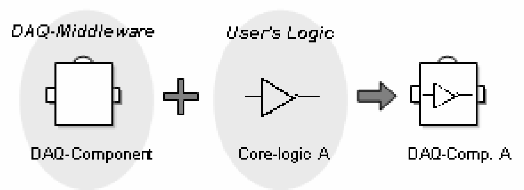

Fig. 2. DAQ-Component and its core-logic.

The features of DAQ-Components include flexibility, scalability and reusability:

- DAQ-Components are independent of each other. Users can choose DAQ-Components, mixing and matching them to create their own DAQ systems.
- The data transfer between DAQ-Components is

location-transparent. This feature enhances scalability of performance. If the beam intensity increases and computers running many DAQ-Components have a heavy workload, we can move some of them to another computer for load sharing.

- DAQ-Components are reusable. Users do not need to develop their requisite DAQ-Components if they already exist.

2. 3 System Configuration

We have utilized XML documents to describe the DAQ system configuration. These documents are stored in a file, called the configuration file. XML has the following advantages: (a) It is independent of the computer environment (b), it is expandable, and (c) it has many software tools and utilities. All information on the DAQ-Components is described in the file, including data path information and the parameters of each DAQ-Component.

2. 4 System interface protocol

We chose XML/HTTP as the interface protocol between DAQ-Middleware and other systems. XML/HTTP has the following advantages:

- It is a simple and well known protocol.
- It is associated with various types of client software, such as Web browsers.
- It enables display of visual images.
- It has affinity with WEB technologies.

MLF also chose the same protocol as DAQ-Middleware to communicate between the user interface and software components such as "Experimental control", including the DAQ sub-system [9]. Therefore, DAQ-Middleware does not require any alterations of interface protocol to build it into a DAQ sub-system of MLF.

3. Implementation of DAQ-Components for a DAQ Sub-system at MLF

We modified the existing DAQ-Components for day one experiments at MLF. This included modifications of the core-logic of two DAQ-Components, the Gatherer and the Monitor.

3. 1 Gatherer Component

A push model communication scheme was implemented for the Gatherer. In the push model, read-out modules initiate data transfer to the Gatherer. The data transfer protocol of NEUNET modules, however, adopts the pull model. In the pull model, the Gatherer needs to request data from NEUNET modules. We therefore modified the core-logic of the Gatherer. We have also implemented other alterations:

- The Gatherer buffers data and creates packets whose sizes are specified by the configuration file. The Gatherer sends the packets to the next DAQ-Component.
- The Gatherer adds a header and footer to the data received from a NEUNET module.

The header and footer data added by the Gatherer guarantee the validity of the data contained in the packet. In addition, the fixed size of packet data makes it easy to debug transferred data.

3. 2 Monitor Component

The Monitor uses ROOT [9], which was developed at CERN. MLF will use the Manyo Library [10], a data analysis software framework for neutron experiments. We developed the core-logic of the Monitor using the Manyo Library and a gnuplot.

4. Beam tests

In September 2007, we performed beam tests to evaluate DAQ-Middleware. We used a thermal neutron beam at the MUSASI beam port of the JRR-3 research reactor, located at the Japan Atomic Energy Agency (JAEA). Fig. 3 shows the schematics of the DAQ system for the beam tests. We used forty PSD detectors and five NEUNET modules. To validate data obtained by DAQ-Middleware, we utilized data obtained using the DAQ system, which is previously used at KENS [1].

The aims of the beam test were to determine if:

- The Gatherer and the Logger could read data from five NEUNET modules concurrently and store their data to files.

- The Monitor could analyze the data on-line and display the results.

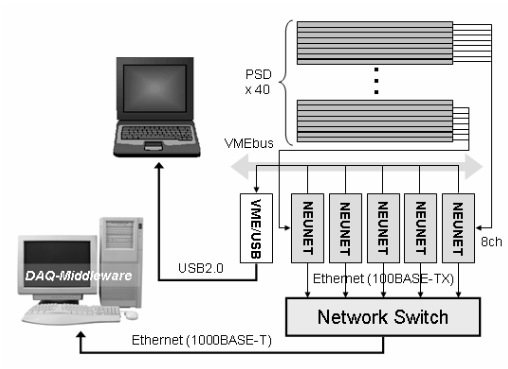

Fig. 3. Schematics of the DAQ setup for the beam-test.

The beam-test showed that both requirements were met. Fig. 4 shows one of the data sets obtained by the beam tests with DAQ-Middleware. The left figure is a picture of PSD detectors with the characters “KEK!” using cadmium plates. The right figure shows 2-D imaging data obtained by the DAQ-Middleware.

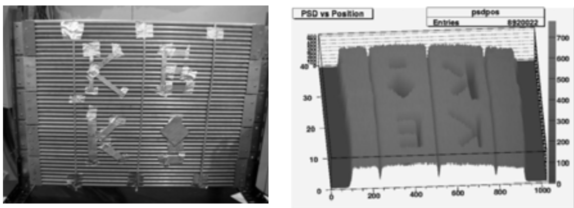

Fig.4. 2-D imaging data obtained by forty PSD detectors. The left figure is a picture of PSD detectors with the characters “KEK!” using cadmium plates. The right figure shows 2-D imaging data obtained by a neutron beam scattered by a polyethylene block.

5. Summary

We describe here the development of the DAQ sub-system of MLF at J-PARC using fundamental DAQ-Components in DAQ-Middleware. We modified two of the fundamental DAQ-Components, the Gatherer and the Monitor, enabling them to be applied to the DAQ sub-system at MLF. Results of beam tests using the DAQ-Components in September 2007 certified the required fundamental DAQ functions of DAQ-Middleware.

Acknowledgements

The authors wish to thank the close collaboration with N. Ando, T. Kotoku and S. Hirano of National Institute of Advanced Industrial Science and Technology. This work is done in the framework of the KEK Detector Technology Project (KEKDTP) of KEK, Japan.

References

[1] S. Satoh, T. Ino, M. Furusaka, Y. Kiyanagi, N. Sakamoto, and K. Sakai, "Development of readout systems for neutron detectors", Nuclear Inst. Meth. A. **529, 421-424, 2004.**

[2] T. Uchida and M. Tnaka, "Development of TCP/IP processing hardware," IEEE Nuclear Science Symposium 2006 Conference Record, pp.1411-1414.

[3] T. Uchida, "Hardware-based TCP processor for gigabit ethernet,"IEEE Trans. Nucl. Sci., 55, 1631-1637, 2008

[4] S. Satoh, S. Muto, N. Kaneko, T. Uchida, M. Tanaka, Y. Yasu, K. Nakayoshi, E. Inoue, H. Sendai, T. Nakatani, and T. Otomo, "Development of a readout system using high-speed network for J-PARC", presented at the International Syposium on Pulsed Neutron and Muon Sciences (IPS08), March 2008, Mito(Japan).

[5] Y. Yasu, K. Nakayoshi, E. Inoue, H. Sendai, H. Fujii, N. Ando, T. Kotoku, S. Hirano, T. Kubota, and T. Ohkawa, "A data acquisition middleware", IEEE NPSS Real-Time 2007 Conference Record, pp.1-3.

[6] Y. Yasu, E. Inoue, K. Nakayoshi, H. Fujii, Y. Igarashi, H. Kodama, N. Ando, T. Kotoku, T. Suehiro, S. Hirano, "Feasibility of data acquisition middleware based on robot technology", International Conference on Computing in High Energy and Nuclear Physics (CHEP06), 2006, T.I.F.R. Mumbai, India.

[7] N. Ando, T. Suehiro, K. Kitagaki, T. Kotoku, and W. Yoon, "RT-component object model in RT-middleware - distributed component middleware for RT (Robot Technology) -", 2005 IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA2005), p.We-B2-5, 2005.06, Espoo, Finland.

[8] T. Nakatani, T. Aoyagi, T. Otomo, J. Suzuki, Y. Yasu, and M. Arai, "Day-1 status of the software framework for experimental instruments in J-PARC/MLF", presented at International Syposium on Pulsed Neutron and Muon Sciences (IPS08), March 2008, Mito(Japan).

[9] ROOT: An Object-Oriented Data Analysis Framework, http://root.cern.ch/.

[10] J. Suzuki, et al. , "Object-oriented data analysis environment for neutron scattering", Nucl. Instr. Meth Phys. Res. A 534, 175–179, 2004.