Validation of the MONARC Simulation Tools

Y. Morita[1] for the MONARC Collaboration

[1] KEK Computing Research Center, 1-1 Oho, Tsukuba, Ibaraki, Japan

Abstract

The objective of the MONARC project is to identify baseline computing models that could provide viable solutions meeting the data analysis needs of LHC experiments. A powerful and flexible set of simulation tools has been developed to model the performance of distributed computing resources for a set of reconstruction and analysis tasks. In this paper we report the validation of the simulation tools using the testbed environments with Objectivity/DB over LAN and WAN connections.

Keywords: MONARC, Regional Center, Performance Simulation, Objectivity, ODBMS

1 Introduction

The goal of the MONARC project [1] is to identify baseline computing models that could provide viable and cost-effective solutions for the data analysis of the LHC experiments. There are four working groups in the collaboration: (1) the Site and Networks Architecture Working Group, (2) the Analysis Process Design Working Group, (3) the Simulation Working Group and (4) the Testbeds Working Group. A powerful and flexible set of simulation tools has been developed by the Simulation WG to model and to evaluate the performance of the distributed computing resources [2].

The simulation tools are based on Java, and the resource utilization such as CPU, disk and network are calculated with a technique called "process oriented discrete event simulation". The detail of the simulation tools has been described in reference 2.

Evaluation of the computer and network system is the iteration of the processes such as measurement, modeling of the system behavior, development of the simulation tools and the validation of the simulation techniques (Fig. 1) [3]. With sufficient iterations of the above cycle, one could predict the behavior of the system for various types of loads with enough accuracy. Therefore the validation of the MONARC Simulation tools should be closely related to the "level of detail" as the project aims for greater accuracy with greater details of the system modeling.

To validate the MONARC simulation tool, following aspects of modeling are of particular concern:

- sharing of the CPU and I/O resource
- queuing mechanism
- performance of the ODBMS with network
- sharing of the network bandwidth

Of these, A. Dorokhov has compared the behavior of the simulation tool to the queuing model. A good agreement was obtained with the analytical calculations of the model [4].

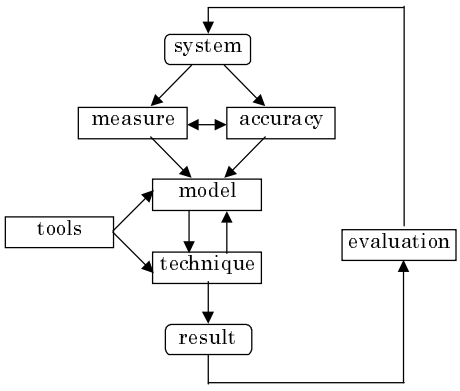

Fig. 1 Performance Evaluation Cycle [3].

2 Testbed Measurements

A several testbed environments have been set up in the Testbeds Working Group in CERN, KEK, INFN, Caltech, and SLAC. These sites are connected with various types of wide area networks, such as dedicated satellite ATM PVC circuits and QoS services. Several example HEP analysis applications, which utilize Objectivity/DB as ODBMS, have been developed and tested over these environments.

To test the performance of Objectivity/DB in a client-server mode over LAN, following set of machines were initially used at CERN [5].

- **machine** A: SUN Enterprise 450 (400MHz x 4 CPUs), 512 MB, two 9 GB disks
- **machine** B: SUN Ultra5 (270MHz CPU), 192 MB memory (Objectivity/DB lock server)
- **machine** C: SUN Enterprise450 (300 MHz dual CPU), 512 MB memory, RAID Disks

All machines are running SUN Solaris 2.6 with C++ v.4.2 and Objectivity/DB v.5.1.

MonteCarlo simulated ATLAS raw data is converted into Objectivity/DB database. A simple C++ program was written to read every object in the event using iterator. Multiple jobs were run on the system with three configurations: (1) Local file database access on machine A, (2) Local file database access on machine C, (3) machine C as an Objectivity/DB AMS server, machine A as a client.

Job execution time and CPU utilization was measured with UNIX *time* command. Local disk I/O speed is measured by a simple C program with *read()* and *write()* system calls. Network speed between machine A and C is measured by FTP. Results of the measurements are given in Table 1 [5].

	machine A local			machine C local			machine C and A		
jobs	time (s)	rms	cpu%	time (s)	rms	cpu%	time (s)	rms	cpu%
1	14.23	0.00	99.1	19.93	0.00	94.5	22.08	0.00	69.7
2	14.44	0.13	196.9	21.04	0.09	181.8	23.43	0.06	131.4
4	14.62	0.05	390.6	38.81	1.95	197.6	30.63	0.79	199.3
8	27.96	1.48	412.0	77.48	2.38	198.0	42.40	1.12	300.1
16	56.59	2.06	407.4	154.41	4.23	199.1	77.50	2.99	332.6
32	114.33	3.57	404.8	309.23	20.08	199.1	151.59	14.57	332.6

Table 1 Job execution time and CPU utilization

3 Comparison with Simulation

To simulate a behavior of a given testbed application, CPU cycles per event, CPU SI95, size of event data, disk I/O speed, network speed must be known to the simulation tools.

By making a simple assumption, how to extract these variables from the testbed measurements has been established. These variables are extracted from a single job accessing local file database (cases (1) and (2) above).

(1) machine A local access - 1 job = 14.23 sec, CPU% = 99.1%
CPU : 17.4 SPECint95 Disk Read : 207 MB/s

(2) machine C local access - 1 job = 19.93 sec, CPU% = 94.5%
CPU : 13.05 SPECint95 Disk Read: 31 MB/s

Suppose the job wall-clock time T_{job} can be divided into $T_{job} = T_{diskread} + T_{process}$, then the ratio of $T_{process}$ and $T_{diskread}$ are the inverse of the CPU ratings and the disk I/O speeds:

$T_{process}$(machine A) / $T_{process}$(machine C) = 13.05 / 17.4

$T_{diskread}$(machine A) / $T_{diskread}$(machine C) = 31 / 207

Under this assumption, $T_{process}$ are calculated as 14.06 sec (machine A) and 18.74 sec

(machine C) respectively, and the CPU utilization 99.0% and 94.0% reproduces the measured value.

By using this set of single job parameters, the simulation tool reproduces the multiple job configurations for the local database access very well. This is an indication that the simulation tool handles the concurrent access of CPU and database file properly. Although the machines A and C are 4- and 2-CPU SMP machines respectively, it has been demonstrated that these machines act like a CPU farm connected with high speed networks for this testbed application.

In Objectivity/DB 5.1, application layer handshaking with page size is observed [6]. For the AMS client/server configuration (3), by introducing network latency per page as a unit of transaction, again the simulation tool reproduces the measurements quite well (Fig. 2).

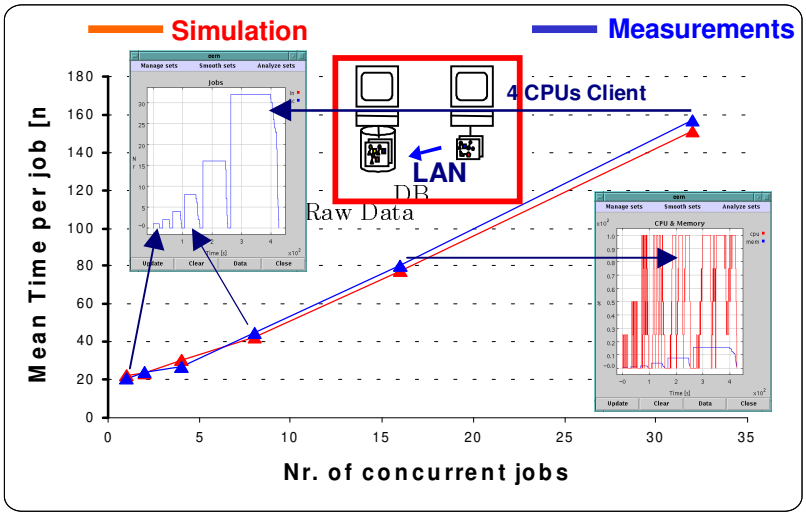

Fig. 2 Simulated and measured job execution time in Objectivity client and server configuration

Although the simulation parameters are not tuned for individual jobs, simulated qualitative behavior of the job execution time looks very similar to the measurements (Fig. xx). This agreement indicates that the time sharing of the CPU resources with multiple jobs as well as the database I/O queuing in the system is well modeled and simulated.

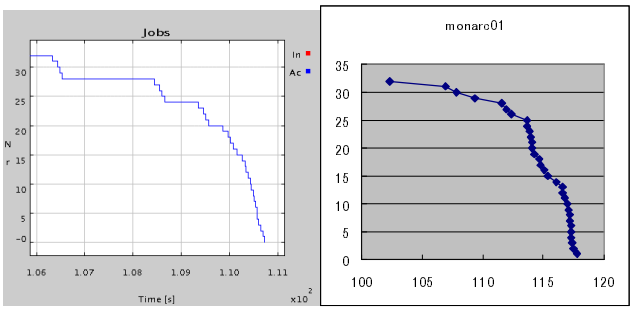

Fig. 3 Simulated and measured distribution of job execution time

4 Other Testbed Measurements

H. Sato has measured the packet-level behavior of Objectivity/DB AMS and DRO (Database Replication Option) with 2Mbps satellite link between CERN and Japan [6]. It was realized that a further optimization is needed to utilize high-bandwidth, long-latency network effectively. Some suggestions for the performance improvements were made to Objectivity.

Italian member of the Testbed Working Group has made an extensive study of Objectivity/DB AMS transaction over WAN [7]. Their measurements were also well reproduced with the simulation tool by I. Legrand [8]. It was also noted that the special care should be taken for the AMS server when high-speed client and low speed client co-exist on the same AMS server.

5 Conclusions

The MONARC project has developed a powerful set of simulation tools to evaluate the performance of the complex computing and network systems, which consist of CPU farms, database servers and local and wide area networks. It has been demonstrated that the Java based program using a process oriented discrete event simulation technique reproduces the predictions of analytical queuing models, as well as a complex set of transactions of ODBMS with local and wide area networks.

However, it is important to understand that the evaluation of the system performance is a continuous cycle of refining the modeling, testing and validation. To make a reliable prediction of the system performance at the startup time of LHC era, careful considerations of the future software performance predictions as well as hardware technology tracking are necessary. An effort of simulating baseline LHC computing models has already been started in the project [9].

Evaluation of the hierarchical mass storage system requires a set of performance measurements with a combination of ODBMS. How to achieve an effective use of ODBMS on tape robotics with various possible object reclustering strategies based on the real HEP data analysis use cases, will be one of the major focus for the next phase of the project [10][11].

Effective use of the wide area network is also one of the most crucial aspect in the LHC era computing. Relatively large latency (several hundreds of milliseconds in round trip time) typically seen in world-wide WAN and in satellite connections, combined with a high bandwidth (several hundreds of Mbps to a few Gbps) foreseen for the future, imposes a serious inefficiency problem for the current implementation of TCP/IP applications. An improved algorithm of TCP/IP slow-start has appeared in the literature [12]. Also several efforts are now underway to implement better implementations of network applications, such as gsiftp. Developments of LHC era world-wide software should be network latency-savvy, and the MONARC project should continue to test and validate the models of networked HEP data analysis with the new generations of software.

References

1 MONARC Project: http://www.cern.ch/MONARC/

2 I. Legrand, "Multi-threaded, discrete event, simulation of distributed computing systems", to be presented in this conference.
3 B. R. Haverkort, "Performance of Computer Communication Systems", John Wiley & Sons Ltd.
4 A. Dorokhov, "Simulation simple models and comparison with queueing theory", MONARC-99/8: http://www.cern.ch/MONARC/docs/monarc_docs.html
5 K. Amako et al., "MONARC testbed and a preliminary measurement on Objectivity AMS server", MONARC-99/7: http://www.cern.ch/MONARC/docs/monarc_docs.html
6 H. Sato, "Evaluation of Objectivity/AMS on the Wide Area Network", to be presented in this conference.
7 A. Brunengo et al., "WAN Test-bed with Objectivity 5.2 in a multi-server configuration", to be presented in this conference.
8 I. Legrand, LCB Marseille Workshop, 1999 September
9 I. Gaines, "Modeling LHC Regional Computing Centers with the MONARC Simulation Tools", to be presented in this conference.
10 K. Holtman, "Data clustering research in CMS", to be presented in this conference.
11 MONARC Phase 3 Letter of Intent: http://www.cern.ch/MONARC/docs/phase3_loi.doc
12 Y. Nishida, J. Murai, Computer Software, Vol. 16, No. 4 (1999) 33.