CDF/DOC/COMP_UPG/PUBLIC/4941

14 NODE TEST OF THE PROTOTYPE FARM

Jaroslav Antos Antonio Wong Chan Paoti Chang
Yen-Chu Chen Stephen Wolbers G. P. Yeh
Ping Yeh

April 5, 1999

1 INTRODUCTION

In mid-October 98, the Run II prototype farm was made available to its main users, CDF, D0 and Theory group. This prototype farm is made up of 14 worker nodes (fnpc100-113) and 4 I/O nodes (fnpca, fnpcb, fnpcc, fnpcd), each containing dual Pentium-II 400 MHz CPU's. In the standard configuration, CDF has exclusive usage of 4 worker nodes (fnpc100-103) and 1 I/O node (fnpcc), although it is possible for one group to use the entire prototype farm for a limited time under certain circumstances. Further details about the prototype farm can be found elsewhere [1,2].

Art Kreymer installed CDF Run II software on the CDF I/O node (fnpcc) in late October 98, and the Production Farm group started to use the prototype farm shortly thereafter. Using FBS (Farm Batch System), a software designed for farm architecture by the Computing Division [3], the Production Farm group has been running Run II Production Executables on the prototype farm since November 98.

In February 99, the Production Farm group performed throughput and reliability tests of the prototype farms using all 14 nodes during a 3-day period, in order to gauge the performance of the prototype farm, to understand the limitations of the prototype farm hardware, and to help us design and optimize the production control software.

In this note, the Production Farm group describes the tests performed and summarizes the results obtained with the 14 node test.

2 GOALS OF THE 14-NODE TEST

The goals of the 14-node tests were to study throughput and reliability of the prototype farm using Run II software.

To perform the tests, the 14 worker nodes were divided into the following queues:

```
fnpc100-103  --> Worker_CDF_1
fnpc104-107  --> Worker_CDF_2
fnpc108-111  --> Worker_CDF_3
fnpc112-113  --> Worker_CDF_4
```

The Production Farm group performed various throughput tests by copying large files ($\sim$ 320 MB) back and forth between the worker nodes and the I/O node via "rcp" and "scp" in a variety of ways.

The reliability tests were done by running Run II Production Executables on the prototype farm for extended periods of time (2-8 hours). The input file for the Production Executable was a Monte Carlo simulation file created with Run II simulation code. Due to time restrictions, it was not possible to run the Production Executables for longer periods of time (1-7 days), although this may be scheduled in future tests. All 14 nodes are not required to make such tests, and they can be attempted at any time.

3 14-NODE TESTS

Below is the list of tests that the Production Farm group performed during the 14 node test period.

- Test 1 - this consisted of copying simultaneously 1 to 24 ($\sim$ 320 MB) files from one disk on the I/O node to the nodes on Worker_CDF_1 and then back from the nodes on Worker_CDF_1 to another disk on the I/O node. All copying was done via rcp, and the rcp processes were initiated on the worker nodes.

- Test 2 - the same as Test 1, but this was done on the nodes on Worker_CDF_2 and Worker_CDF_3.
- Test 3 - the same as Test 1, but this was done on all nodes in the 4 queues (Worker_CDF_1, Worker_CDF_2, Worker_CDF_3 and Worker_CDF_4).
- Test 4 - the same as Test 3, but this was done with "scp", instead of "rcp".
- Test 5 - the same as Test 1, but the copying between I/O node and the nodes on Worker_CDF_1 was divided among 3 disks on the I/O node.
- Test 6 - this consisted of running 23 Production Executable processes simultaneously on all 14 worker nodes, which included reading the input file from I/O node to worker nodes, event reconstruction and copying the output from worker nodes to I/O node. The 23 processes do not represent a limitation on the batch system or Run II software, but was a result of a set-up "bug", which prevented a larger number of processes from running. All I/O was done via rcp.
- Test 7 - same as Test 6, but with 46 Production Executables processes on all 14 worker nodes. The first 23 were submitted, and the remaining 23 sat in the queues until the first 23 finished. All I/O was done via rcp.

4 TESTS RESULTS

Test 1 consisted of 1-24 copying processes from one disk (/cdf/data1/cdfprod0) on the I/O node to the scratch area ($PC_BATCH_SCRATCH) on the worker nodes. After the initial copying was finished, each process copied back the file from the scratch area to /cdf/data2/cdfprod0 on the I/O node. The copying was done using the input file for the CDF Production Executable ($\sim 320MB$ in size).

Table 1 shows the average rate per process, so the total rate is $(rate) \times (\#\ of\ proc)$. The highest rate obtained was 13.3 MB/s for 4 processes running simultaneously. The column under %CPU indicates the percentage of CPU devoted to the copying process, averaged over the copying from I/O node to worker node and vice-versa.

Table 1: Summary of test 1

# of proc.	Job ID	Rate/proc.	%CPU
1	4310	7.40 MB/s	46.4%
2	4312	5.61 MB/s	39.1%
4	4314	3.32 MB/s	31.3%
8	4302	1.24 MB/s	16.5%
16	4304	0.54 MB/s	10.0%
24	4308	0.33 MB/s	6.5%

Test 2 was similar to Test 1, except that nodes on Worker_CDF_2 and Worker_CDF_3 were used instead of Worker_CDF_1. The nodes on these two queues seemed somewhat slower than the nodes on Worker_CDF_1 (see Table 2). More tests need to be done to rule out fluctuations due to usage of the system by other people at the same time.

Table 2: Summary of test 2

# of proc.	Job ID	Rate/proc.	%CPU	Queue
4	4419	2.50 MB/s	18.5%	Worker_CDF_2
4	4321	2.54 MB/s	20.1%	Worker_CDF_3

Test 3 was similar to Test 1, but all nodes on the 4 queues are used this time. The results are summarized in Table 3. The * indicates where "rcp time out problems" occurred, which means the "rcp" copying process aborted before the file copying process was finished. There seems to be a hard limit on how many processes are allowed to run on the I/O node simultaneously (between 28-56). It is not known if the "rcp" problem is due to too many processes on the I/O node or due to an intrinsic "rcp" limitation. We will continue to investigate and attempt to formulate a solution to this problem. For this test, not all "rcp" processes run simultaneously, so one cannot infer from this table the total rate by simply multiplying the rate/proc. by the # of proc.

Test 4 consisted of using standard "scp" (not the faster "blowfish" ver-

Table 3: Summary of test 3

# of proc.	Job ID	Rate/proc.	%CPU
1-1-1-1	4335	5.66 MB/s	32.5%
4-4-4-2	4371	1.10 MB/s	9.1%
6-6-6-6	4341	0.75 MB/s	7.9%
8-8-8-4	4377	0.40 MB/s	9.6% *
12-12-12-6	4383	0.25 MB/s	6.6% *
12-12-12-12	4365	0.20 MB/s	5.9% *

sion) for the throughput test. Our results in Table 4 indicate that "scp" is about 3 times slower than "rcp". However, there were NO time-out problems with "scp".

Table 4: Summary of test 4

# of proc.	Job ID	Rate/proc.	%CPU
8-8-8-4	4423	0.16 MB/s	0.49%

Test 5 was similar to Test 1, except that the input files to be copied to the scratch areas of the worker nodes were divided among 3 disks on the I/O node (/cdf/data1/cdfprod0, /cdf/data2/cdfprod0 and /home/cdfprod0). The process of copying back to the I/O node was also divided among the 3 disks. When the load was divided (a), both "rcp" processes were equally fast, but if the load was put on a single disk (b), then the latter "rcp" process was several times slower. This was due to all "rcp" processes trying to access the same disk at the same time, reaching the limit of the disk (or SCSI bus) I/O rate very quickly. When the load was divided, the average transfer rate was ~ 20.4 MB/s. The results are summarized in Table 5, where %CPU (in) and %CPU (out) refer to the percentage of CPU used by each copying process averaged over 8 processes for copying from the I/O node to the worker node (in) and vice-versa (out).

Table 6 lists the times when the various processes in Job ID 4874 started and ended. Although the jobs started sequentially, the overlap was consider-

Table 5: Summary of test 5

# of proc.	Job ID	Rate(in)/proc.	Rate(out)/proc.	%CPU (in)	%CPU (out)
8 (a)	4874	2.38 MB/s	2.73 MB/s	16.2%	15.4%
8 (b)	4880	3.95 MB/s	0.70 MB/s	48.5%	3.2%

able, so our throughput rates were a good indicator of the true performance of the system.

Table 6: Summary of test 5

# of proc.	Job ID	Section #	Time started	Time ended
2	4874	test1	17:12:24	17:15:47
2	4874	test2	17:13:37	17:18:16
2	4874	test3	17:14:53	17:19:50
2	4874	test4	17:16:01	17:21:52

Test 6 was a test of the reliability of the prototype farm. The total CPU time for each process was $\sim 9,930$ sec on average, which includes the time to copy all necessary files from the I/O node to the worker node, to run the Production Executable on the worker node and to copy the output files from the worker nodes to the I/O node. Each process took $\sim 10,502$ sec of clock time on average. The clock time includes the time needed by FBS to schedule, start/end a job, to copy files from the I/O node to the worker node (and vice-versa) and to run the Production Executable. From these two values, we obtained an average CPU utilization rate of 94.6%. Table 7 summarizes all the results, including the %CPU used by the I/O processes and %CPU used by the Production Executable.

On the matter of event processing rate and memory usage, the first Production Executable run on the prototype farm in mid-November 98 took about 7.0 CPU sec/event and used about 18.6 MB of memory on the worker nodes. In late January 99, this became 7.8 CPU sec/event and 25 MB of memory, respectively. The Production Executable used for the 14 node test

Table 7: Summary of test 6

		I/O		Production Executable		
proc. #	Job ID	Rate/proc.	%CPU	CPU time/event	Clock time/event	% CPU
1	4389	4.24 MB/s	12.56%	9.87 sec.	10.12 sec	98.2%
2	4389	3.28 MB/s	12.86%	9.81 sec.	10.07 sec	98.2%
3	4389	4.55 MB/s	13.10%	9.78 sec.	10.06 sec	98.0%
4	4389	4.72 MB/s	14.48%	9.84 sec.	10.16 sec	97.6%
5	4389	4.32 MB/s	24.94%	9.88 sec.	10.22 sec	97.3%
6	4389	4.33 MB/s	24.90%	9.84 sec.	10.18 sec	97.3%
7	4389	5.04 MB/s	25.46%	9.85 sec.	10.23 sec	97.0%
8	4389	3.59 MB/s	24.78%	9.90 sec.	10.30 sec	96.9%
9	4389	3.32 MB/s	12.66%	9.81 sec.	10.20 sec	96.8%
10	4389	3.12 MB/s	13.62%	9.81 sec.	10.24 sec	96.5%
11	4389	3.45 MB/s	13.61%	9.80 sec.	10.24 sec	96.4%
12	4389	3.70 MB/s	14.30%	9.77 sec.	10.22 sec	96.3%
13	4389	5.25 MB/s	25.54%	9.80 sec.	10.33 sec	95.5%
14	4389	3.97 MB/s	19.74%	9.81 sec.	10.34 sec	95.6%
15	4389	4.67 MB/s	15.74%	9.84 sec.	10.42 sec	95.1%
16	4389	4.04 MB/s	16.06%	9.71 sec.	10.23 sec	95.7%
17	4389	3.97 MB/s	13.52%	9.86 sec.	10.40 sec	95.5%
18	4389	3.37 MB/s	12.28%	9.79 sec.	10.37 sec	95.1%
19	4389	4.05 MB/s	13.32%	9.85 sec.	10.53 sec	94.3%
20	4389	3.74 MB/s	13.20%	9.74 sec.	10.29 sec	95.4%
21	4389	4.60 MB/s	24.56%	9.79 sec.	10.45 sec	94.4%
22	4389	4.61 MB/s	24.84%	9.79 sec.	10.51 sec	93.8%
23	4389	4.24 MB/s	19.78%	9.77 sec.	10.47 sec	94.1%

took 9.8 CPU sec to process an event and used about 46 MB of memory. These two parameters are growing quickly, and it may have an impact on the specifications of the worker nodes for the Run II farms. This will have to be closely monitored from now on.

Test 7 was similar to Test 6, except that 2 consecutive 23-process jobs were submitted on the 14 worker nodes. The rates shown in Table 8 are the average over each of the 23-process jobs and are comparable to the individual rates shown in Table 7. These two consecutive 23-processes jobs on the 14 worker nodes took about 8 hours to complete. The system was able to sustain the continuous processing with an average CPU utilization rate of $\sim 95.1\%$. The addition of CDF catalog, calibration, trigger database accesses, etc is likely to lower the CPU utilization rate. The impact of these tasks on the CPU utilization rate should be minimized by carefully designing the access mechanisms.

Table 8: Summary of test 7

		I/O		Production Executable		
# of proc.	Job ID	Rate/proc.	%CPU	CPU time/event	Clock time/event	% CPU
23	4441	4.35 MB/s	20.59%	9.86 sec.	10.12 sec	96.5%
23	4559	3.94 MB/s	18.69%	9.87 sec.	10.26 sec	96.8%

5 CONCLUSIONS

- The Production Farm group has learned a lot with the 14-node tests, specially on the throughput tests.
- The data transfer rate between the I/O node and the worker nodes in a realistic scenario is ~ 20 MB/s. Note that the current prototype farm is not our final choice of hardware standards for the I/O node or switches. We are encouraged by the results obtained, and we believe that a different system using newer technology will allow us to achieve higher rates comfortably.

- Our average CPU utilization rate when running the CDF Production Executable on the prototype farm is $\sim 95\%$, including the time needed by FBS to schedule, start and end a process.

- The Production Farm project is ramping up. The development of the Production Farm software will require frequent interactions with the rest of the CDF Offline (CDF Catalog, Calibration files, Tape I/O, Simulation, etc) and Computing Division.

6 FUTURE PLANS

1) The Production Farm group plans to start accessing Calibration, Trigger Database, CDF Catalog, etc when the necessary hooks or access points become available. We expect some of these hooks to become available beginning with the May release of the Run II code.

2) Another aspect of our testing of the prototype farm will involve tape I/O. Our current estimate is that we will start our "Tape I/O integration" in mid-99.

3) Our specification of the worker nodes will be made soon, and we expect delivery to start by October 99, with installation and general availability sometime later. One aspect of the specification is the memory size for each processor. Our current specifications call for 256 MB/processor, and the ability to upgrade it by a limited factor.

4) The Production Farm group is exploring plans to offer the prototype farm as a small production facility for Monte Carlo. This could help the Production Farm group to test production-like conditions, to establish possible techniques for simulating events on a large scale and to test production control software.

5) We plan to test our production control code for splitting/merging of output data in the near future. This is dependent on first defining the output streams on Run II code and defining a mechanism by which the production control software can access this information. There are still on-going discussions among the Offline group members as to when this will occur.

7 REFERENCES

[1] Jaroslav Antos, et al., CDF Production for Run II, CDF note 4810 (1998).

[2] Mark Breitung, et al., Processing Farms plans for CDF and D0 for Run II, proceedings of CHEP98 (http://www-hpcc.fnal.gov/farms/papers/chep98.ps), Chicago, IL (1998).

[3] Mark Breitung, et al., Farm Batch System (FBS) Users Guide (http:://www-hpcc.fnal.gov/farms/r2farms/users_guide/users_guide.html) Chicago, IL (1998).