ASC Student Supercomputer Challenge (2015)

Preliminary Contest Notifications

Dear Participating Teams:

Thank you very much for participating in the ASC Students Supercomputer Challenge 2015 (ASC15). This document will provide detailed information about the preliminary round of the contest.

1. About the Preliminary Round

In the preliminary round, each registered team is required to submit a set of documents that include a proposal, optimized source code files and output files (detailed requirements specified in the Appendix A). The proposal needs to be written in English, and will be reviewed by the ASC evaluation committee.

2. Submission Guideline

All teams should make their submissions to info@asc-events.org before 9:00 AM, March 19th, 2015 (UTC/GMT +8：00). The confirmation of your submission will be sent to you by email. The submission should include the following items:

a) The proposal file (in .doc or .pdf format), named with the university or college name and the contact person name (e.g. AAAUniversity_BBB.doc).

b) The additional files should be compressed into one file (e.g. AAAUniversity_BBB.zip, other compression formats are also OK). The compressed file should at least include:

- Output files of HPCC
- Log files of NAMD
- Optimized source codes and log file of Gridding

3. For any further inquiries about the contest, please contact the ASC committee through the following emails:

a) Technical support: techSupport@asc-events.org

b) Contest organization: info@asc-events.org

c) News and media: media@asc-events.org

We wish you the best of luck at ASC15

ASC15 committee

2015-1-13

Appendix A: Proposal Requirements

I. A brief background description of the university`s or the department`s supercomputing activities (5 points)

1. Supercomputing-related hardware and software platforms
2. Supercomputing-related courses, trainings, and interest groups
3. Supercomputing-related research and applications
4. A detailed description of the key achievements on supercomputing research (no more than 2 items), attached with proof materials (published papers, award certificates, etc.)

II. Introduction of the Team (5 points)

1. Brief description about for the process of your team-up
2. Brief introduction of each team member (including group photos of the team)
3. Your team slogan.
4. Your team Logo (JPG, high resolustion, 300DPI)

III. Technical proposal requirements (90 points)

1. Design of your HPC system (15 points)

a) Within the 3,000-watt power budget, your system should be designed to achieve the best performance of HPCC and the NAMD application.

b) In this part, you need to specify your system's software and hardware configuration, interconnection, to describe the power consumption and performance evaluation, and to provide your analysis on the advantages and disadvantages of the proposed architecture.

c) Your system should be based on the Inspur NF5280M4 server. The components listed in the table below will be provided by Inspur to the teams that enter the final. Other components (except the server itself) are acceptable, but should be prepared by the teams at their own costs. For example, you can change the number of NF5280M4 servers and accelerators, the type of the hard disk and memory, and even the type of the Ethernet in your proposed configuration.

Item	Name	Configuration
Server	Inspur NF5280M4	CPU: Intel Xeon E5-2670v3 x 2， 2.3Ghz， 12 cores Memory: 16G x8， DDR4， 2133Mhz Hard disk: 300G SAS x 1

		Power consumption estimation: *E5-2680v3 TDP 115W,memory nominal 7.5W, hard disk nominal 10W*
Accelerator card	XEON PHI-31S1P	Intel XEON PHI-31S1P（57 cores, 1.1GHz, 1003GFlops, 8GB GDDR5 Memory） *Power consumption estimation: 270W*
HCA card	FDR	Infiniband Mellanox ConnectX®-3 HCA card, single port QSFP, FDR IB *Power consumption estimation:9W*
Switch	GbE switch	10/100/1000Mb/s，24 ports Ethernet switch *Power consumption estimation:30W*
	FDR-IB switch	SwitchX™ FDR InfiniBand switch, 36 QSFP port *Power consumption estimation:130W*
Cable	Gigabit CAT6 cables	CAT6 copper cable, blue, 3m
	Infiniband cable	Infiniband FDR optical fiber cable, QSFP port, cooperating with the Infiniband switch for use

2. HPCC Test (15 points)

The team shall run the HPCC test on their own hardware. The target is to achieve the highest efficiency within the 3KW power budget.

The proposal should include descriptions of the software environment (operating system, complier, math library, MPI software and HPCC version, etc.), the testing method, performance optimization methods, performance estimation, problem and solution analysis, etc. In-depth analysis into HPCC's algorithm and source code is highly encouraged.

The HPCC software can be downloaded from: http://icl.cs.utk.edu/hpcc/software/index.html Successful verification and optimization of HPCC on hardware platforms will be given additional scores. However, teams without the required hardware platforms are also encouraged to submit their thoughts and analysis.

Appendix C is suggested to be used for submitting HPCC results.

3. The NAMD Test (20 points)

Similar to the HPCC test, the team shall run the NAMD test on their own hardware to verify the correctness and to achieve good performance and efficiency.

The proposal document shall include the description of the testing software environment (operating systems, compliers, math libraries, MPI software and application software, etc. with version information), the testing methods, performance optimization methods, performance estimation, problem and solution analysis, etc. In-depth analysis into NAMD's algorithm and source code is highly encouraged.

The NAMD software can be downloaded from: http://www.ks.uiuc.edu/Research/namd/.
As NAMD provides different workloads that demonstrate different performance features, in this contest, the teams should focus on the two NAMD workloads that can be downloaded from http://www.ks.uiuc.edu/Research/namd/utilities/f1atpase.tar.gz and http://www.ks.uiuc.edu/Research/namd/utilities/apoa1.tar.gz.
However, teams without the required hardware platforms are also encouraged to submit their thoughts and analysis.

4. Optimization of the Gridding program on the CPU+MIC Platform (40 points)

a) Application background

The teams are required to conduct performance optimization of the Gridding program based on the CPU+MIC heterogeneous platform.

The Square Kilometre Array (SKA) project is the largest radio telescope projects in human history, in which the world's largest telescope will be built. In the project, to reconstruct the image of the sky from the telescope data, we need to go through a series of processing, including correlating, Gridding, Fast Fourier Transform (FFT) and so on.

In the Gridding program, the visibility data items that are non-uniformly sampled at discrete locations in u, v space will be convolved with a convolution function that has a finite-size kernel. After the convolution, the resulting data are placed on a rectangular grid. The processing is called gridding. It scales with the number of visibility samples, and the size of the convolutional kernel.

Key Words: Gridding, Square Kilometre Array, radio telescope, imaging process, convolution function

b) Introduction to application program

i.Program download link: asc15.inspur.com:/home/Gridding (on the remote testing platform)

The entire package includes:

- tConvolve.tar: the Gridding program source code
- verify.tar: the program for performing result verification

ii.The Gridding program compilation and testing instructions:

- Compilation command: make
- Running command: ./tConvolve
- Test cases:

Workload	Input files
Workload1	input.dat, randnum.dat

- Output file1: Array “grid” stores the computing results which are outputted in a file named “grid.dat” that will be used in the verification program to verify the correctness of the code.
- Output file2: The Gridding time and the Gridding rate will be outputted in a log file named “log.dat”.

iii.The Verification program compilation and testing instructions:

- Compilation command: make
- Running command: ./verify
- Before running the verification code, the “grid.dat” file should be copied into the same directory.
- The "grid_std.dat" file contains the computing results of the original Gridding program.
- Requirements for result correctness: relative error of the L1 norm should be smaller than 1e-12.

c) Requirements for preliminary contest

i. The proposal should include a section to describe the design and optimization of the Gridding program on the CPU+MIC platform. The section should include description of the software and hardware environment, parallelization design methods, performance optimization methods, testing process and results, etc. In-depth analysis regarding principles, parallel algorithm and codes are recommended.

ii. After parallelization and optimization, the final program should be tested with the Workload1 case on **one** computing node in the CPU+MIC hybrid cluster (the remote testing platform) provided by the organization committee. The CPU+MIC hybrid cluster consists of three computing nodes. The performance analysis in your proposal should be based on the results of this test. Each team is required to pack and submit your optimized source code and the log file of this test to ASC committee.

d) Specific notes

i. ASC committee provides remote testing platform that includes CPU and MIC cards. The detailed information of the testing platform is provided in "Appendix B. Description of the remote testing platform". Hardware configuration of the platform is fixed.

ii. The original code is written in C++. The revised code should include appropriate comments.

iii. You can run your program either on CPUs or on both CPUs and MICs. You may use some parallel programming methods such as MPI, OpenMP, pThread and OpenCL to write your code, but all these methods should be supported by MIC. If the program utilizes the computing resources of the MIC cards, you must use the MIC offload programming model.

iv. The function named "gridKernel" is the kernel function, which is the only function that is required to be optimized. The value of the "Gridding rate: (million grid points per second)", which will be outputted in the log file, is the only indicator of the performance. All the modifications of the program should be within the region that we measure the running time. There are two timers at the beginning and the end of this region respectively. The difference between the values of these two timers is the running time of the kernel, which will be used to compute the Gridding rate.

v. The optimization methods must not violate the basic convolution algorithm that is used in the original code.

vi. The optimization methods that are used in the kernel function must be independent of the initial conditions.

This application will also be tested in the final contest, but on a larger CPU+MIC hybrid cluster.

Appendix B. The Remote Testing Platform

I. Configuration

The platform consists of 14 nodes: 7 login/compiling nodes and 7 computing nodes. The detailed configuration is as follows:

- 6 login/compiling nodes for login, compiling CPU application and submitting multi-node jobs to the computing nodes. These nodes can be accessed through the same IP address, and the users will be automatically distributed into different nodes;

Item	Name	Configuration
Server	Inspur NF5280M4 x 6	CPU：Intel Xeon E5-2680v3 x 2, 2.5Ghz, 12 cores
		Memory: 16G x8, DDR4, 2133Mhz
		Hard disk: 300G SAS x 1

- 1 node for compiling MIC application.

Item	Name	Configuration
Server	Inspur NF5280M3 x 1	CPU：Intel Xeon E5-2692v2 x 2, 2.2Ghz, 12 cores
		Memory: 16G x8, DDR3, 1600Mhz
		Hard disk: 300G SAS x 1
		Accelerator card: Intel XEON PHI-31S1P （57 cores, 1.1GHz, 1003GFlops, 8GB GDDR5 Memory） x1

- A small-scale CPU cluster with 4 computing nodes.

Item	Name	Configuration
Server	Inspur NF5280M4 x 4	CPU：Intel Xeon E5-2680v3 x 2, 2.5Ghz, 12 cores
		Memory: 16G x8, DDR4, 2133Mhz
		Hard disk: 300G SAS x 1

- A small-scale CPU+MIC hybrid cluster with 3 computing nodes.

Item	Name	Configuration
Server	Inspur NF5280M3 x 3	CPU：Intel Xeon E5-2692v2 x 2, 2.2Ghz, 12 cores
		Memory: 16G x8, DDR3, 1600Mhz
		Hard disk: 300G SAS x 1
		Accelerator card: Intel XEON PHI-31S1P （57 cores, 1.1GHz, 1073.6GFlops, 8GB GDDR5 Memory） x 1

II. Instructions of remote testing platform

1. Login account:

Login IP address: asc15.inspur.com (by ssh)

The login account and password will be notified to each team through E-mail. Users shall change the password after first login. The login account will be closed automatically after the final contest. One account allows at most 2 users to log in simultaneously.

2. Storage space:

Each team will have 20GB storage space by default. Users shall make their own data backup and clear the disk space timely. In case the user needs to use a larger disk space, please contact techsupport@asc-events.org.

3. Usage guidelines

As mentioned above, the login/compiling nodes shall be used for login, compiling, and doing single-node tests (note that doing single-node tests could influence other user's logining). Multi-node test should be performed on the computing nodes by submitting jobs.

For more detailed description, please refer to "ASC15 Student Supercomputer Challenge remote testing platform manual", which will be sent out with the account.

For any other related questions, please also contact techsupport@asc-events.org

As requested by the Beijing government, the platform has to be shutdown from Feb.17 to Feb.25 during China's traditional holiday Spring Festival.

Appendix C. HPCC results submission form

HPCC	
HPCC version	
System Spec	
Compute nodes	
OS	
MPI	
Compiler	
Compiler Flags	
Environment variables:	
Turbo (ON/ OFF)	

	Results
Number of nodes	
Number of cores (MPI Tasks)	
HPCC (base or optimized)	
HPL_N	
HPL_NB	
HPL_npcol	
HPL_nprow	
HPL_Tflops	
PTRANS_nb	
PTRANS_GBs	
StarDGEMM_Gflops	
SingleDGEMM_Gflops	
StarSTREAM_Copy	
StarSTREAM_Scale	
StarSTREAM_Add	
StarSTREAM_Triad	
SingleSTREAM_Copy	
SingleSTREAM_Scale	
SingleSTREAM_Add	
SingleSTREAM_Triad	
MPIRandomAccess_GUPs	
StarRandomAccess_GUPs	
SingleRandomAccess_GUPs	
MPIRandomAccess_LCG_GUPs	
StarRandomAccess_LCG_GUPs	
SingleRandomAccess_LCG_GUPs	
MPIFFT_Gflops	

StarFFT_Gflops	
SingleFFT_Gflops	
MaxPingPongLatency_usec	
AvgPingPongLatency_usec	
MinPingPongLatency_usec	
MaxPingPongBandwidth_GBytes	
AvgPingPongBandwidth_GBytes	
MinPingPongBandwidth_GBytes	
RandomlyOrderedRingLatency_usec	
RandomlyOrderedRingBandwidth_GBytes	
NaturallyOrderedRingLatency_usec	
NaturallyOrderedRingBandwidth_GBytes	