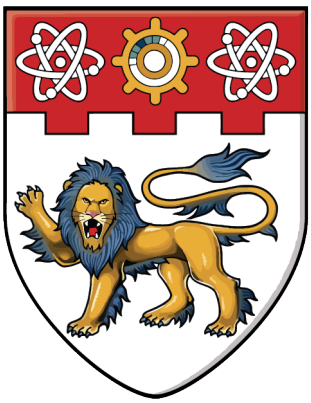

NANYANG TECHNOLOGICAL UNIVERSITY

SC19 STUDENT CLUSTER COMPETITION

ARCHITECTURE PROPOSAL

Submitted By: Team Supernova
Team Supervisor: Prof. Lee Bu Sung, Francis

1 Hardware configuration

Table 1: Hardware configuration

Item	Spec	Quantity
Servers [0-1]	Supermicro SuperServer 4029GP-TRT	2
Servers [2-3]	HPE Apollo 6500 Gen 10	2
Ethernet Switch	Netgear GS324	1
Infiniband Switch	Mellanox EDR Switch	1
CPU [0-1]	Dual Intel Xeon Platinum 8160 (24C, 2.1GHz, 3.7GHz Turbo, 33MB L3, 150W)	
CPU [2-3]	Dual Intel Xeon Platinum 8260 (24C, 2.4GHz, 3.9GHz Turbo, 35.75MB L3, 165W)	
GPU	NVIDIA Tesla V100	16
RAM [0-1]	256 GB DDR4	
RAM [2-3]	192 GB DDR4	
Storage (Boot)	2.96 TB across 4 SATA SSDs	
Storage (Auxiliary shared data)	1TB NVMe SSD	
Storage (Parallel cluster filesystem)	9.4 TB across 7 NVMe SSDs	

We see a lot of potential in using GPU to accelerate some of the applications like VPIC and benchmarks for SC19. While for other applications like SST, they are more CPU intensive. Thus, we need to strike a balance between the needs of different applications. We will bring 16 GPUs to the competition site, but the actual number of GPUs to be used will only be determined on site via benchmarks with the PDU readings, so that we do not exceed the 3k Watts power limit.

2 Software used

Table 2: Software Configuration

Item	Spec
Operating System	Oracle Linux 7
Toolchains	Intel & GNU
MPI Libraries	MVAPICH, Intel MPI & OpenMPI (with UCX)
Profilers	ARM Forge & ARM Performance Reports
Environment Management	Lmod
Shared File System	NFSv4
Cluster File System	BeeGFS
System Monitoring	Grafana

2.1 Operating System

Orale Linux 7 is another RHEL derivative, similar to CentOS. It is stable and suitable for computational tasks, and its userspace is compatible with the vendor packages we wish to install. Despite us using CentOS for previous competitions, we are comfortable and confident in switching to Oracle Linux this year due to our previous experience with CentOS.

2.2 Toolchains

We are utilizing both the Intel and GNU toolchains. Intel compilers excel at improving performance by utilizing multiple cores and employing vector instructions, but may not implement certain GNU extensions that could be required by applications. GNU compilers implement those required extensions and can hence produce executables for applications utilizing them, allowing us to cope with a variety of different applications. GNU compilers can also provide more GPU-friendly executable files as well as more stable performance over different code source.

2.3 MPI Libraries

We have tested applications with different MPI libraries and in turn chose the most suitable ones for this competition. The Intel MPI library is used for applications requiring high intra-node (shared memory) communication performance, while the MVAPICH and OpenMPI libraries are used for their excellent multi-rail Infiniband support to boost performance of applications that require high inter-node bandwidth. GPU applications are also specifically linked with OpenMPI, due to its CUDA-aware nature and support for GPU-specific features like GPUDirect RDMA and NVIDIA gdrcopy, helping us accelerate the performance of GPU applications.

2.4 Profilers

Our team employed ARM performance reports to obtain overviews of application performance, and then the ARM Forge profiler to help us narrow down and investigate specific regions of poor performance. This lets us efficiently fine tune our applications so as to achieve the best performance with our limited compute resources.

2.5 Other Software

We have also installed many supporting tools to increase the efficiency and effectiveness of managing the cluster and running applications. We are utilizing NFSv4 to share auxiliary data among different nodes, and the high-performance BeeGFS to store both input and output data for applications. To monitor the status of the cluster as a whole, we employ Grafana.