International Conference
"Distributed Computing and Grid-Technologies in Science and Education"

June 30 – July 5, 2014, Dubna, Russia

GRID'2014

Book of Abstracts

JOINT INSTITUTE FOR NUCLEAR RESEARCH
LABORATORY OF INFORMATION TECHNOLOGIES

DISTRIBUTED COMPUTING AND GRID-TECHNOLOGIES IN SCIENCE AND EDUCATION

Book of Abstracts of the 6th International Conference

Dubna, June 30 – July 5, 2014

РАСПРЕДЕЛЕННЫЕ ВЫЧИСЛЕНИЯ И ГРИД-ТЕХНОЛОГИИ В НАУКЕ И ОБРАЗОВАНИИ

Тезисы докладов 6-й Международной конференции

Дубна, 30 июня – 5 июля 2014 г.

Dubna 2014

INTEGRATION OF THE COMPUTING CLUSTER INTO THE INFORMATION SYSTEM OF FACILITY

KULYABOV Dmitry, GEVORKYAN Migran, SEVASTYANOV Leonid
yamadharma@gmail.com, mngevorkyan@sci.pfu.edu.ru, mngevorkyan@sci.pfu.edu.ru,
Peoples' Friendship University of Russia

The experience of integration of the computing cluster into the information system of facility is described. The cluster is comprised of one control node (2xCPU Intel Xeon E5-2670, 64 Гб RAM) and two compute nodes (lxCPU Xeon E5-2670, 64 Гб RAM, 1xGPU Nvidia Tesla M2090).

The following problems had to be solved during cluster configuration.

Integration of the cluster authentication system with faculty's local network authentication system.

Setting the consistent software updates on the cluster's control node and the compute nodes.

The centralized initial cluster setup with help of xcat utility.

Installing additional software with CUDA technology support.

Creating the local software repository for specific software support (for example, PBS / Torque with CUDA support).

The following software were to be installed on the cluster: GNU Compiler Collection (gcc) [2], with gfortran version 4.4.7 c OpenMP [3], Julia language compiler [4], Python interpreter version 2.7. Oslo following libraries have been installed: OpenMPI [5], MPICH [6], BLAS [7], LAPACK [8], GSL (GNU Scientific Library) [9], GAMESS (US) (General Atomic and Molecular Electronic Structure System) [10], Python NumPy and SciPy [11].

All the software listed above is widely known and a good account of itself in the field of scientific computing. However, it may be useful to give a little bit more details about Julia programming language — the new open source programming language that is in the stage of active development. Julia language has created to achieve following goals: multiparadigmality, focus on scientific computing, the possibility of interactive language constructions execution (like iPython), achieving the performance close to C/C++ and Fortran, built-in support of parallel computing, easy integration of an external Fortran and C/C++ libraries.

Литература

1. Геворкян М.Н., Королькова А.В., Кулябов Д.С. Настройка высокопроизводительного вычислительного комплекса // // Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологичных систем / РУДН. — Москва: РУДН, 2014. — С. 171.
2. GCC, the GNU Compiler Collection — http://gcc.gnu.org/
3. OpenMP API specification for parallel programming — http://openmp.org/wp/
4. The Julia Language — http://julialang.org/
5. Open MPI: Open Source High Performance Computing — http://www.open-mpi.org/
6. MPICH: High-Performance Portable MPI — http://www.mpich.org/