

romeoLAB : HPC Training Platform on HPC facility

JEAN-MATTHIEU ETANCELIN, University of Reims Champagne-Ardenne - CReSTIC EA3804

ARNAUD RENARD, University of Reims Champagne-Ardenne - CReSTIC EA3804 – ROMEO HPC Center

In this pre-exascale era, we are observing a dramatic increase of the necessity of computer science courses dedicated to parallel programming with advanced technologies on hybrid architectures. The full hybrid cluster *Romeo* has been used in that purpose since a long time in order to train master students and cluster users. The main issue for trainees is the cost of accessing and exploiting a production facility in a pedagogic context. The use of some specific techniques and software (SSH, workload manager, remote file system, ...) is mandatory without being part of courses prerequisites nor pedagogic objectives. The romeoLAB platform we developed at ROMEO HPC Center is an online interactive pedagogic platform for HPC technologies courses. Its main purpose is to simplify the process of resources usage in order to focus on the taught subjects. This paper presents the context, the romeoLAB architecture as well as its motivations, usages and pedagogical contents.

KEYWORDS

HPC, Teaching in HPC, Education, Tools for CSE/HPC/DA Education, Programming Environments/Languages for CSE/HPC/DA Education, Infrastructures for CSE/HPC/DA Education, Experiences in CSE/HPC/DA Education

1 ROMEO HPC CENTER CONTEXT

The ROMEO HPC Center - Grand-Est region, France - is a High Performance Computing (HPC) platform hosted by the University of Reims Champagne-Ardenne (URCA) since 2002. Its goal is to provide high performance computing resources for both local industrial and academic researchers, along with an entire ecosystem of services like secured storage space, scientific software, in-depth expertise in scientific application domains, visualization tools, scientific awareness actions and innovation in architecture: since 2013, our cluster *Romeo* is the French most powerful hybrid CPU-GPU supercomputer, ranked 151th in the TOP500 list.

One of the ROMEO HPC Center's missions is to lead some educational activities related to HPC, GPU and hybrid computing. ROMEO is involved in the management of the 2-years Master curriculum dedicated to High Performance Computing and Simulation. In 2015, ROMEO received a grant from Intel Corporation to enhance teaching in manycore computations. Since 2016, the ROMEO HPC Center is also awarded as a GPU Education Center by Nvidia for supporting the educational activities in the specific field of GPU technologies. This label rewards and supports trainings and master degree courses organized by ROMEO HPC Center through the romeoLAB platform.

2 romeoLAB MOTIVATION AND FEATURES

We build romeoLAB with four main requirements. First of all, we need to execute code on a real HPC facility, because designing efficient parallel algorithms must be evaluated on real hardware architectures to experiment hardware related phenomenon like deadlocks, bandwidth bottleneck or cache misses. Furthermore, our cluster *Romeo* is equipped with all modern hardware resources we need for multiple-nodes labs, and we can rely on the ROMEO HPC system engineer to install needed software. Unfortunately, usual tools for exploiting a high performance computing cluster are rarely part of prerequisite and pedagogic objectives of courses. In a traditional training, students and participants have to use, among others, an ssh client, the cluster workload manager and remote file systems. An introduction part dedicated to these tools is therefore mandatory and time consuming. These are obstacles to the pedagogical process efficiency and

consists in the most important motivation for the romeoLAB development. Furthermore, compatibility of all the different client tools and server-side service is always an issue and we don't want lab sessions to become support sessions for student heterogeneous laptop configuration.

That's why, as a second requirement, we want our solution to be accessible without any client software except a recent web-browser that everyone already has. This aspect is important because ROMEO organize trainings on participant site, where computer and networks are generally out of our control, especially in the context of international collaborations with universities worldwide (Africa, Asia, ...). Furthermore, a web interface can provide high-level pedagogic contents while hiding difficult access to HPC facility. This web interface is responsible for managing available sessions and labs for each user, retrieving labs content from the versioning system (we use GIT), and starting labs on the cluster via the batch scheduler (we use Slurm). Identified teachers can also update courses content via the same web interface.

Our third requirement is to completely integrate development environments (edition, compilation and execution from web-based interface) into pedagogical content (texts, figures, PDF, videos, ...) to create a modern MOOCs (Massive Open Online Courses) solution. For being a standard and possible to integrate it into our platform with plug-ins, we choose the *Jupyter Notebook* as the main solution. It provides an interactive interface for data science and scientific computing that propose, in the same web page that contains the multimedia pedagogical content, to edit code via advanced editor with syntax coloration, to compile and execute codes thru bash cells and to dynamically create content, like, for example, figures based on execution results. *Jupyter* make it also possible to generate slides from current notebook, download all the contents and export notebook for attendees. Everything is executing on the compute node -in a protected context- directly available from internet thanks to an in-house and secured web proxy. Cell execution is also possible directly through the workload manager, that enable to mutualize large resources (for a 16 node MPI course by example), between dozens of students.

Sometimes, a web interface is not enough. For example, developers may need to run graphical software or GUI to profile their code with NVCC, TAU or PGProf. Other scientific users want to start their application like Fluent, or Paraview. That is our fourth requirement, and we integrate the ability to reach a VNC Linux graphical desktop on the compute nodes through a websockified *noVNC* client. VNC servers are launched on user demand, at run time.

3 RESULTS AND AVAILABLE CONTENT IN romeoLAB

Our platform has been successfully used in various contexts since march 2016, with a number of participants varying from 10 to 60. All these usages have been performed in an instructor-led course but it also can be used in a self-led format without any restrictions: GPU programming for JDEV, advanced Python for HPC, Profiler Days with a TAU, MAQAO and SCALASCA, OpenFOAM School, Master Courses, Python for Science at LoOPS, MPI&OpenACC session at GTCEU2016, GPU Spring School, and advanced multi-GPU tools at GTC2016.

romeoLAB is now addressing a wide range of technologies and audience levels. The idea is to mutualize pedagogic efforts at ROMEO scale and to host external courses. We encourage teachers of the University to collaborate on courses through this platform for sharing experiences, cluster hardware and software.

romeoLAB is intended to be improved, to receive new applications –with or without teaching purpose- and to be distributed to others HPC Centers who showed their interest after public sessions or presentations.