Channabasaveshwara Institute of Technology
(Affiliated to VTU, Belgaum & Approved by AICTE, New Delhi)
(NAAC Accredited & ISO 9001:2015 Certified Institution)
NH 206 (B.H. Road), Gubbi, Tumkur – 572 216. Karnataka.

organizes

1st International Conference on Innovation in Computer Science, Electrical and Electronics Engineering
(ICICEE-2020)

In association with

Technical Institute For Engineers

Own HPC Cluster Based on Virtual Operating System

Tomasz RAK Rzeszow University of Technology
Łukasz SCHIFFER Rzeszow University of Technology

Own HPC Cluster Based on Virtual Operating System

Tomasz RAK Łukasz SCHIFFER

Department of Computer and Control Engineering
Rzeszow University of Technology
POLAND

RZESZOW UNIVERSITY OF TECHNOLOGY

1st International Conference on Innovation in Computer Science, Electrical and Electronics Engineering
ICICEE-2020

Tomasz RAK, Łukasz SCHIFFER Own HPC Cluster Based on Virtual... 1/18

Table of contents

- Introduction (problem)
- Clusters (HPC)
- Construction of the Environment (virtual cluster)
- Tests (performance)

Table of contents

- Introduction – Motivation, a problem statement and my approach
- Clusters
- Construction of the Environment
- Tests

Approaches

(AWS ParallelCluster, OpenHPC, Virtual HPC Cluster on Docker container, The Rugged Cloud vHPC Cluster)

Distributed computing allows to share tasks between nodes.

Types of clusters:

- High Availability also known as Failover Clusters
- Load Balancing
- High Performance

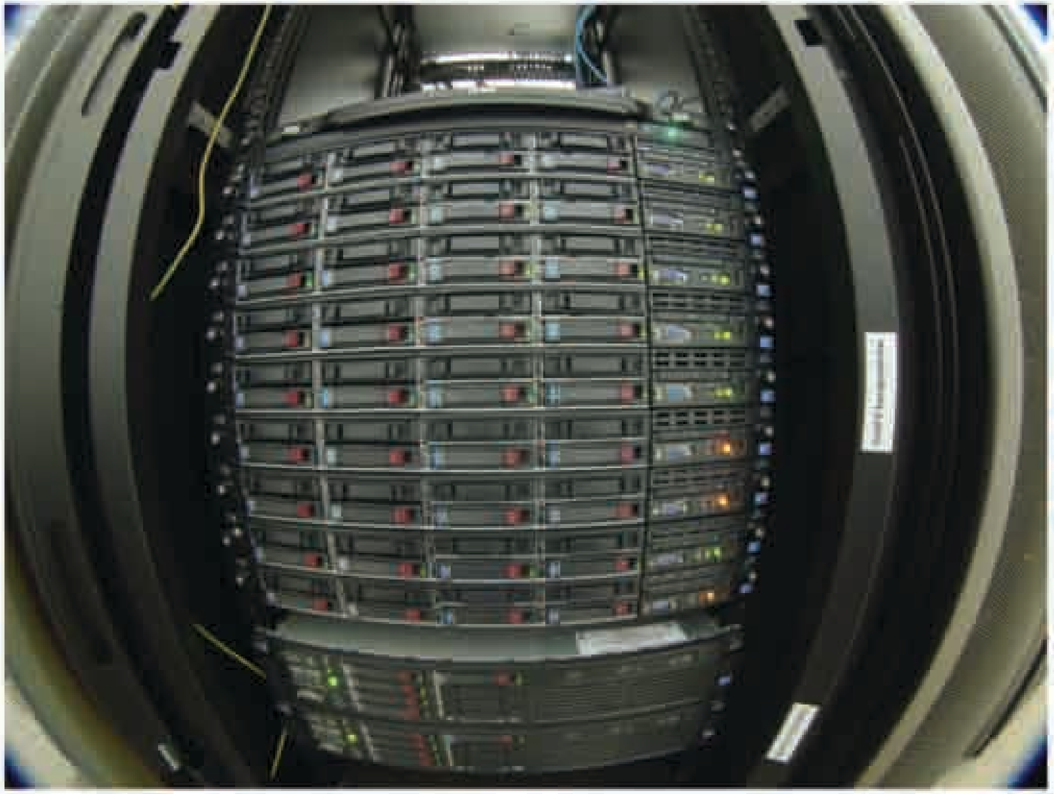
Our works

- Preparation of a virtual environment based on VirtualBox
- Launch of several virtual machines with the PelicanHPC Linux operating system
- Automatic configuration of the HPC cluster
- Analysis of sample results of applications running on the cluster

Table of contents

- Introduction
- Clusters – `Multi-node structure`
- Construction of the Environment
- Tests

Real Cluster vs. Virtual Cluster



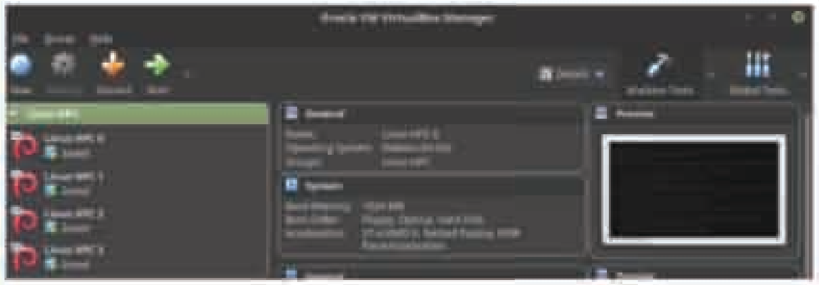


Table of contents

- Introduction
- Cluster
- Construction of the Environment – `Virtual machine`
- Tests

Types of hypervisors

- First one:
  - VMware ESXi
  - Microsoft Hyper-V
  - KVM
- Second one:
  - VMware
  - Oracle VM VirtualBox
  - QEMU

GNU/Linux distribution PelicanHPC

Debian Live project

It is focused on MPI-based parallel computing on a cluster.

ISO images

All of the nodes of the cluster get their file systems from the same image, so it is guaranteed that all nodes run the same software.

MPI Library

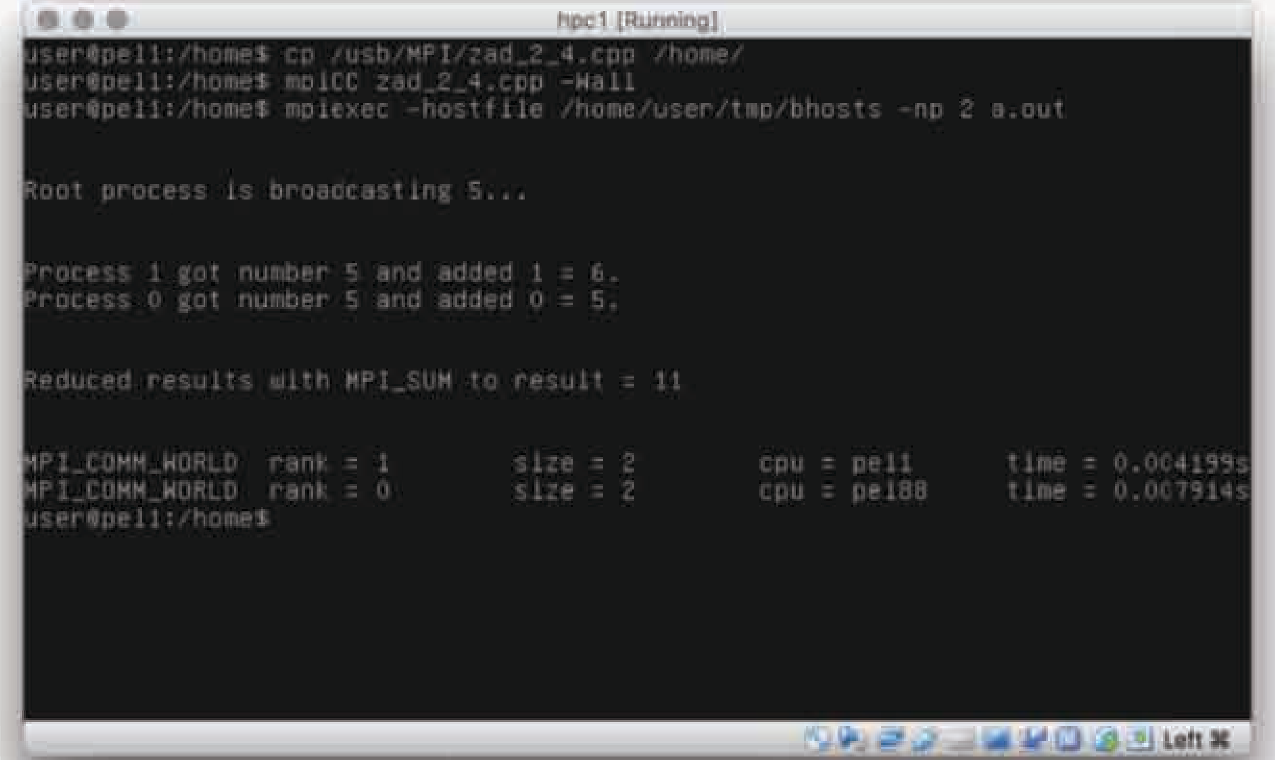
MPI is therefore a standard library for performing parallel processing using a distributed memory model.

- MPICH2
- Intel MPI
- HP MPI
- Open MPI is one implementation of the MPI specification

Root node and several computation nodes

- Virtual Root Node Configuration
- Running Root Node
- Configuring Persistent Storage Shared within Cluster
- Troubleshooting Problem with SSH Config File
- Connecting to Virtual Machine via SSH
- Virtual Computational Nodes Configuration
- Configuring and Running Virtual Cluster
- Managing Virtual Cluster when Not in Use

PelicanHPC



```
user@pel1:/home$ cp /usb/MPI/zad_2_4.cpp /home/
user@pel1:/home$ mpiCC zad_2_4.cpp -Wall
user@pel1:/home$ mpiexec -hostfile /home/user/tmp/bhosts -np 2 a.out


Root process is broadcasting 5...


Process 1 got number 5 and added 1 = 6.
Process 0 got number 5 and added 0 = 5.


Reduced results with MPI_SUM to result = 11


MPI_COMM_WORLD  rank = 1        size = 2        cpu = pel1      time = 0.004199s
MPI_COMM_WORLD  rank = 0        size = 2        cpu = pel88     time = 0.007914s
user@pel1:/home$
```

Table of contents

- Introduction
- Clusters
- Construction of the Environment
- Tests Experiments

Example program (matrix multiplication)[1]

100 measurements were made for processes {2, 3, 4, 5, 6, 7, 8, 9, 10, 25, 50, 100, 150, 200} and nodes {1, 2, 3} and for MPI library

```
#include "mpi.h" /*...*/
int main (int argc, char *argv[])
{ /*...*/
    std::vector< std::vector< double> > a; /* matrix A to be multiplied */
    std::vector< std::vector< double> > b; /* matrix B to be multiplied */
    std::vector< std::vector< double> > c; /* result matrix C */ /*...*/
    MPI_Init(&argc,&argv);
    MPI_Comm_rank(MPI_COMM_WORLD,&taskid);
    MPI_Comm_size(MPI_COMM_WORLD,&numtasks); /*...*/

/*************************** master task ************************************/ /*...*/
    /* Send matrix data to the worker tasks */ /*...*/
    /* Receive results from worker tasks */ /*...*/

/*************************** worker task ************************************/ /*...*/
    MPI_Recv(&offset, 1, MPI_INT, MASTER, FROM_MASTER, MPI_COMM_WORLD, &status);
    MPI_Recv(&rows, 1, MPI_INT, MASTER, FROM_MASTER, MPI_COMM_WORLD, &status); /*...*/

    MPI_Send(&offset, 1, MPI_INT, MASTER, FROM_WORKER, MPI_COMM_WORLD);
    MPI_Send(&rows, 1, MPI_INT, MASTER, FROM_WORKER, MPI_COMM_WORLD); /*...*/

   MPI_Finalize();
}
```

[1] https://github.com/trak2020z/matrix_multiplication

The comparison (2)

Table: Minimum execution times of the matrix multiplication program

Number of processes	MPI time [s] 1 node	MPI time [s] 2 nodes	MPI time [s] 3 nodes
2	1.0248	1.0489	1.0447
3	0.709	0.5315	0.5428
4	0.701	0.3718	0.369
5	0.657	0.3656	0.2941
6	0.6462	0.3481	0.2523
7	0.6333	0.3828	0.2718
8	0.6356	**0.3423**	0.2687
9	**0.6301**	0.3725	**0.2468**
10	0.6393	0.3458	0.2901
25	0.7833	0.5026	0.3872
50	1.2286	0.8972	0.7281
100	3.0185	2.4396	1.9804
150	5.6363	5.1608	4.0358
200	17.39	9.4426	7.0858

Table: Maximum execution times of the matrix multiplication program

Number of processes	MPI time [s] 1 node	MPI time [s] 2 nodes	MPI time [s] 3 nodes
2	1.0582	25.1577	1.3854
3	1.0489	**9.3603**	21.4408
4	0.718	12.1667	20.165
5	0.7962	11.695	11.6808
6	0.6827	9.8887	**0.2656**
7	0.7317	19.5889	6.4869
8	**0.6663**	22.5477	0.3522
9	0.7177	10.5114	10.4185
10	0.6723	33.3004	9.7081
25	0.8549	11.3014	22.6319
50	2.1687	47.2693	29.4186
100	7.651	43.613	57.7428
150	16.09	73.6401	47.8839
200	73.0198	132.4574	83.2886

Conclusions

Created virtual cluster is ready for real usage

- Creation of virtual cluster is easy and does not need additional computers.
- With operating system dedicated to this task such as PelicanHPC you do not have to worry about problems neither with configuring local network for the cluster, nor installation of the required software.
- It is available to be built by everyone and can be used for home or educational purposes to show how distributed computing works.
- Also as our example shows it can be used for development and testing of several programming approaches to high performance computing, allowing to choose best solution of the problem.
- Later the chosen application can be run on a real, physical cluster fulfilling its purpose.

Own HPC Cluster Based on Virtual Operating System

Thank you for your attention!

Introduction (4)

Clusters (7)

Construction of the Environment (9)

Tests (15)

The European Commission has offered its support in the field of HPC in order to help find a treatment for the novel coronavirus.
Supercomputers help researchers speed drug discovery for Covid-19.