Acta de Reuniones Metavolante #1

Stiven Agudelo
Aurelio Vivas
Miguel Correa

Reunión 1 - Martes 27 de Julio 2021 (19:30 PM - 21:30 PM)

Asistentes

- Stiven Agudelo
- Miguel Correa

Discusión

- ¿Como entramos a cronos?
- ¿El compilador de Intel ya se encuentra instalado?
- ¿Qué sería lo primero que deberíamos hacer para terminar la meta volante?

Conclusiones y decisiones

- El compilador de intel aparentemente no se encuentra instalado, por lo que debemos instalarlo antes de instalar los demás drivers y aplicativos para compilarlos con dicho compilador.

Avances

- Setup de acceso para el cluster con el archivo ~/.ssh/config y el copiado de relaciones de confianza con los nodos y el proxyjump

Tareas

1. Aprender a usar el compilador de intel
2. Configurar la red Infiniband
3. Configurar RDMA
4. Configurar NFS

Reunión 2 - Jueves 29 de Julio 2021 (19:00 PM - 19:30 PM)

Asistentes

- Stiven Agudelo
- Aurelio Vivas

Discusión

- Stiven explicó a Aurelio las actividades que se habían acordado con Miguel.

Conclusiones y decisiones

- Se acordó hacer una próxima reunión para definir la metodología de trabajo.

Tareas

- Programar una siguiente reunión para realizar el plan de trabajo
- Programar los horarios en los que nos vamos a reunir para la realización del plan.

Reunión 3 - Viernes 30 de Julio 2021 (18:30 PM - 20:30 PM)

Asistentes

- Aurelio Vivas
- Stiven Agudelo

Discusión

- Documentar reuniones anteriores.
- Leer HPL y listar lo que se necesita instalar
- Diagrama de despliegue con componentes a instalar
- Ambiente de pruebas en vagrant.
- Conocer la arquitectura del cluster Cronos(arquitectura de memoria, versión software, soporte para instrucciones vectoriales).
- Organización de github (un readme por paso de instalación de componente)
    - Un archivo para blas (pasos de instalación y referencias)
    - Un archivo para hpl
    - Un archivo para infiniband
    - Un archivo para sistema de archivos compartido
- Desarrollo de pruebas en la máquina virtual y realización de pasos
- Despliegue en clúster

Conclusiones y decisiones

- Se decidió usar vagrant para que cada miembro del equipo pueda desplegar un cluster de máquinas virtuales en su computador y de esta forma avanzar en la generación del manual de instrucciones para la instalación de las dependencias de HPL.
- Nos reuniremos el sábado 1 de agosto a las 3 pm.

Avances

- Creación de organización (hpc-cios-grupo5) en github para las metas, creación del repositorio meta 1. Para el desarrollo de la meta volante 1.
- Creación de repositorio con los archivos de Vagrant que serán usados como infraestructura de pruebas.
- Instalación de vagrant y VirtualBox para poder usar vagrant.
- Despliegue del cluster de máquinas virtuales en el ambiente de pruebas de cada miembro del equipo.
- Reconocimiento arquitectura del cluster

```
Architecture:        x86_64
CPU op-mode(s):      32-bit, 64-bit
Byte Order:          Little Endian
CPU(s):              16
On-line CPU(s) list: 0-15
Thread(s) per core:  1
Core(s) per socket:  8
Socket(s):           2
NUMA node(s):        2
Vendor ID:           GenuineIntel
CPU family:          6
Model:               45
Model name:          Intel(R) Xeon(R) CPU E5-2670 0 @ 2.60GHz
Stepping:            6
CPU MHz:             3292.090
BogoMIPS:            5187.41
Virtualization:      VT-x
L1d cache:           32K
L1i cache:           32K
L2 cache:            256K
L3 cache:            20480K
NUMA node0 CPU(s):   0-7
NUMA node1 CPU(s):   8-15
Flags:               fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat
pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb rdtscp lm
constant_tsc arch_perfmon pebs bts rep_good nopl xtopology nonstop_tsc cpuid aperfmperf
pni pclmulqdq dtes64 monitor ds_cpl vmx smx est tm2 ssse3 cx16 xtpr pdcm pcid dca sse4_1
sse4_2 x2apic popcnt tsc_deadline_timer aes xsave avx lahf_lm epb pti ssbd ibrs ibpb
stibp tpr_shadow vnmi flexpriority ept vpid xsaveopt dtherm ida arat pln pts md_clear
flush_l1d
```

- Qué dependencias requiere HPL (documentación de netlib)
    - Compilador de Intel (OneAPI Basic, One2108741463API HPC)
    - Basic Linear Algebra Subprograms BLAS o the Vector Signal Image Processing Library VSIPL
    - MPI
    - Infiniband
    - Sistema de Archivos Compartido (NFS, etc)
- Descarga de los paquetes de OneAPI para instalar el compilador de Intel.

Tareas

- Leer documentación del compilador de C y MPI de la librería OneAPI.
- Instalar OneAPI (Basic y HPC) y documentar los pasos de instalación en un archivo readme en el repositorio de la meta volante 1.
- Descargar los drivers de infiniband y entender las configuraciones posibles para las especificaciones de Cronos, documentar las configuraciones que deberían usarse en Cronos.
- Configurar el sistema de archivos compartidos

Reunión 4 - Sábado 31 de Julio 2021 (15:30 PM - 20:00 PM)

Asistentes

- Stiven Agudelo
- Miguel Correa
- Aurelio Vivas

Discusión

- Decisión de qué drivers a instalar en los nodos

Conclusiones y decisiones

- El compilador de intel aparentemente no se encuentra instalado, por lo que debemos instalarlo antes de instalar los demas drivers y aplicativos para compilarlos con dicho compilador

Avances

- Se descargo usando yum utilidad para saber que targeta de red tiene el nodo

```
yum install pciutils
```

- Reconocimiento de las tarjetas de red en los nodos de cronos

```
07:00.0 Network controller: Mellanox Technologies MT27500 Family [ConnectX-3]
```

- Drivers de Mellanox-OFED
- Instalamos GCC usando Yum y demás paquetes necesarios para el rebuild de los drivers de Mellanox para el kernel que estábamos usando de acuerdo a las instrucciones del output de la instalación del driver usando el comando
-

```
yum install \
    gcc \
    kernel-devel-4.18.0-193.el8.x86_64 \
    gdb-headless \
    rpm-build \
    kernel-rpm-macros \
    elfutils-libelf-devel
```

```
gcc (GCC) 8.4.1 20200928 (Red Hat 8.4.1-1)
Copyright (C) 2018 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

-

Tareas

5. Aprender a usar el compilador de intel
6. Configurar la red Infiniband
7. Configurar RDMA
8. Configurar NFS

https://meet.google.com/tii-hoki-kwb

Reunión 5 - Lunes 2 de Agosto 2021 (19:00 PM - 22:00 PM)

Asistentes

- Stiven Agudelo
- Miguel Correa
- Aurelio Vivas

Discusión

- El driver que estábamos tratando de instalar previamente no estaba funcionando bien por lo que se usó yum

Conclusiones y decisiones

- Decidimos usar los drivers que se usan en Apolo con las mismas tarjetas de infiniband por recomendación de Santigo Gil

Avances

- Se descargo usando yum utilidad para saber que targeta de red tiene el nodo

```
yum install pciutils
```

- Reconocimiento de las tarjetas de red en los nodos de cronos

```
07:00.0 Network controller: Mellanox Technologies MT27500 Family [ConnectX-3]
```

- Drivers de Mellanox-OFED
- Instalamos GCC usando Yum y demás paquetes necesarios para el rebuild de los drivers de Mellanox de para el kernel que estábamos usando de acuerdo a las instrucciones del output de la instalación del driver usando el comando
-

```
yum install \
    gcc \
    kernel-devel-4.18.0-193.el8.x86_64 \
    gdb-headless \
    rpm-build \
    kernel-rpm-macros \
    elfutils-libelf-devel
```

```
gcc (GCC) 8.4.1 20200928 (Red Hat 8.4.1-1)
Copyright (C) 2018 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

Tareas

Intel Integrated Performance primitives Cryptography 2021.3.0
*Intel onePI Threading Building Block 2021.3.0

*Intel Distribution for GDB 10.1.2
*Intel DPC++ Compatibility Tool 2021.3.0
*Intel oneAPI DPC++/C++ Compiler 2021.3.0
*Intel oneAPI DPC++ Library 2021.4.0
Intel oneAPI Data Analytics Library 2021.3.0
Intel oneAPI Deep Neural Network Library 2021.3.0
*Intel oneAPI Math kernel Library 2021.3.0
*Intel Integrated Performance Primitives 2021.3.0
Intel Advisor 2021.3.0
Intel Distribution for Python 2021.3.0
*Intel oneAPI Collective Communications Library
Intel FPGA Add-on for oneAPI Base Toolkit 2021.1.3.0
Intel oneAPI Video Processing Library 2021.4.0
Intel VTune(TM) Profiler

Reunión 6 - Martes 3 de Agosto 2021 (19:00 PM - 22:00 PM)

Asistentes

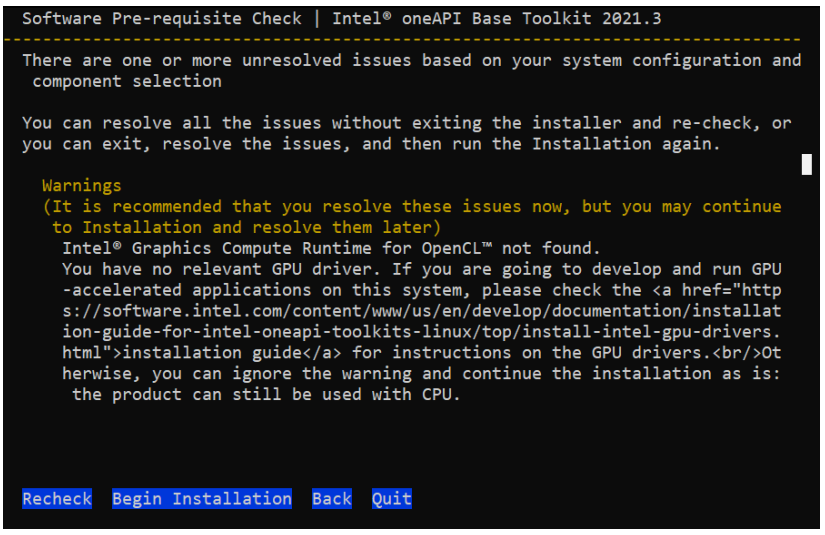
- Stiven Agudelo
- Miguel Correa
- Aurelio Vivas

Discusión

- Instalación de OneAPI y NFS

Conclusiones y decisiones

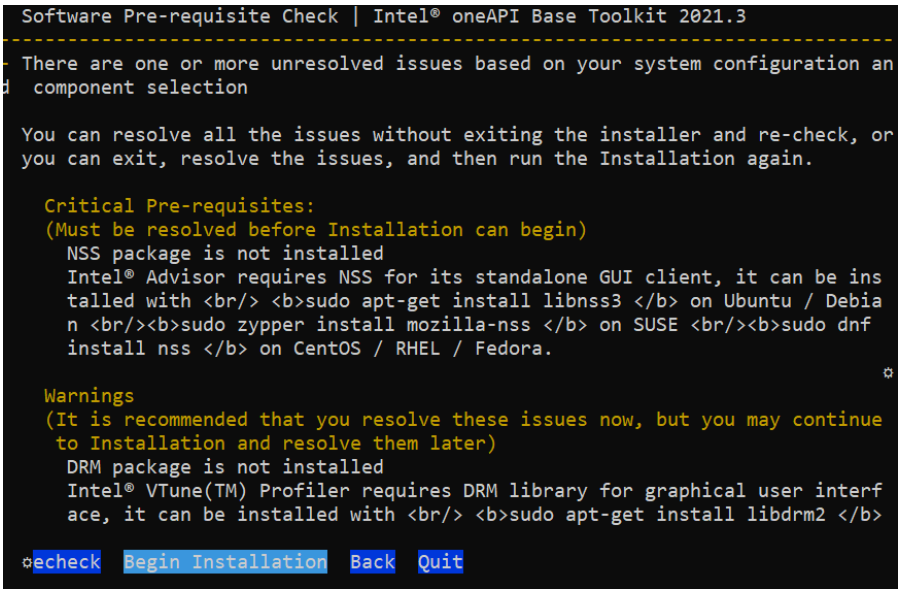
- OneAPI es una herramienta que tiene todo lo que necesitamos para HPL optimizado para arquitectura intel:
    - NPI
    - NKL
    - HPL



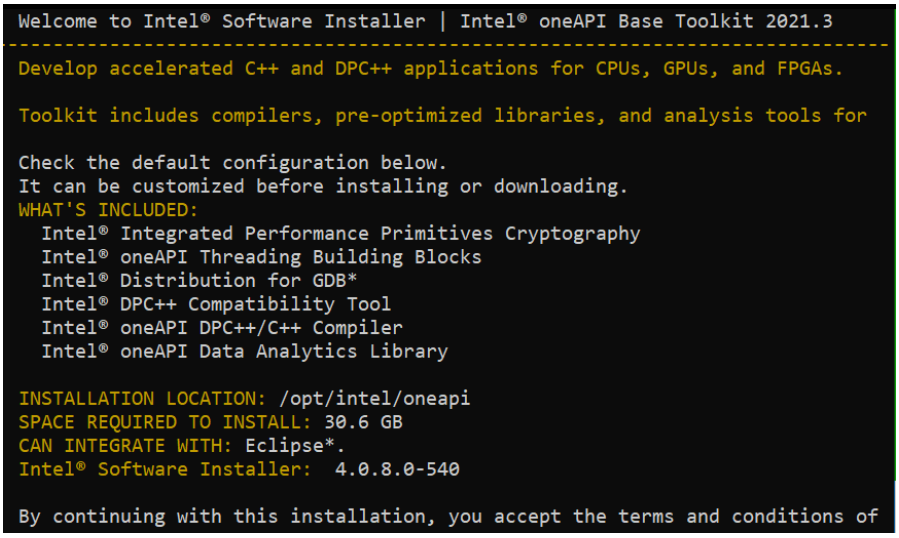
https://software.intel.com/content/www/us/en/develop/documentation/get-started-with-intel-oneapi-base-linux/top/before-you-begin.html?cid=oth&campid=iags_install&source=installer

https://software.intel.com/en-us/forums/intel-oneapi-base-toolkit

https://www.mellanox.com/pdf/prod_software/Red_Hat_Enterprise_Linux_(RHEL)_7.6_Driver_User_Manual.pdf
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/networking_guide/sec-testing_early_infiniband_rdma_operation



sudo dnf install nfs



https://platform9.com/docs/openstack/tutorials-setup-nfs-server

Reunión 7 - Jueves 5 de Agosto 2021 (19:00 PM - 22:00 PM)

Asistentes

- Stiven Agudelo
- Miguel Correa
- Aurelio Vivas

Discusión

- Pruebas de MPI y validación de los dispositivos de red

Conclusiones y decisiones

- El equipo se dio cuenta que la conexión de infiniband no estaba disponible porque faltaba la perfilación de la conexión

```
[mpiuser@compute-1-28 ~]$ mpirun -n 20 -hosts master,slave1 /bin/hostname
[mpiexec@compute-1-28] check_exit_codes
(../../../../../src/pm/i_hydra/libhydra/demux/hydra_demux_poll.c:117): unable to run
bstrap_proxy on master (pid 33557, exit code 768)
[mpiexec@compute-1-28] poll_for_event
(../../../../../src/pm/i_hydra/libhydra/demux/hydra_demux_poll.c:159): check exit codes error
[mpiexec@compute-1-28] HYD_dmx_poll_wait_for_proxy_event
(../../../../../src/pm/i_hydra/libhydra/demux/hydra_demux_poll.c:212): poll for event error
[mpiexec@compute-1-28] HYD_bstrap_setup
(../../../../../src/pm/i_hydra/libhydra/bstrap/src/intel/i_hydra_bstrap.c:1062): error waiting for
event
[mpiexec@compute-1-28] HYD_print_bstrap_setup_error_message
(../../../../../src/pm/i_hydra/mpiexec/intel/i_mpiexec.c:1014): error setting up the bootstrap
proxies
[mpiexec@compute-1-28] Possible reasons:
[mpiexec@compute-1-28] 1. Host is unavailable. Please check that all hosts are available.
[mpiexec@compute-1-28] 2. Cannot launch hydra_bstrap_proxy or it crashed on one of
the hosts. Make sure hydra_bstrap_proxy is available on all hosts and it has right
permissions.
[mpiexec@compute-1-28] 3. Firewall refused connection. Check that enough ports are
allowed in the firewall and specify them with the I_MPI_PORT_RANGE variable.
[mpiexec@compute-1-28] 4. Ssh bootstrap cannot launch processes on remote host.
Make sure that passwordless ssh connection is established across compute hosts.
[mpiexec@compute-1-28]    You may try using -bootstrap option to select alternative
launcher.
[mpiuser@compute-1-28 ~]$
```

El error se solucionó al especificar la interfaz de red a usar
mpirun -n 4 -ppn 2 -iface eno1 -hosts master,slave1 /bin/hostname

Validar el estado de los dispositivos de red. Se muestra únicamente la lista de los dispositivos reconocidos por la Network Manager.
Referencia: RedHat Networking Guide

```
nmcli device status

DEVICE  TYPE        STATE         CONNECTION
eno1    ethernet    connected     System eno1
ib0     infiniband  disconnected  --
eno2    ethernet    unavailable   --
lo      loopback    unmanaged     --
```

Añadir una conexión significa crear un perfil de configuración que es asignado al dispositivo. Por ejemplo, para añadir un perfil de configuración Ethernet con IP dinámica, permitiendo DHCP para asignar la configuración de red.

```
nmcli connection add type ethernet con-name connection-name ifname
interface-name
```

Por ejemplo, para crear una conexión llamada *my-office*

```
nmcli con add type ethernet con-name my-office ifname ens3

Connection 'my-office' (fb157a65-ad32-47ed-858c-102a48e064a2) successfully
added.
```

Para abrir o activar la conexión

```
nmcli con up my-office

Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/5)
```

Para ver información detallada de la conexión nueva

```
nmcli -p con show test-lab

===============================================================================
                    Connection profile details (test-lab)
===============================================================================
connection.id:                          test-lab
connection.uuid:                        05abfd5e-324e-4461-844e-8501ba704773
connection.interface-name:              ens9
connection.type:                        802-3-ethernet
connection.autoconnect:                 yes
```

```
connection.timestamp:                   1410428968
connection.read-only:                   no
connection.permissions:
connection.zone:                        --
connection.master:                      --
connection.slave-type:                  --
connection.secondaries:
connection.gateway-ping-timeout:        0
[output truncated]
```

NetworkManager will set its internal parameter connection.autoconnect to yes. NetworkManager will also write out settings to /etc/sysconfig/network-scripts/ifcfg-my-office where the corresponding BOOTPROTO will be set to none and ONBOOT to yes.

Infiniband Networking RedHat

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/networking_guide/ch-configure_infiniband_and_rdma_networks

Software packages
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/networking_guide/sec-infiniband_and_rdma_related_software_packages

Reunión 8 - Sábado 7 de Agosto 2021 (18:00 PM - 21:00 PM)

Asistentes

- Stiven Agudelo
- Aurelio Vivas

Discusión

- Arreglando problemas de infiniband

Conclusiones y decisiones

- Se configuraron los perfiles de conexión entre ambos nodos.

Comando para obtener la mac address del dispositivo de red, este comando fue utilizado para obtenerla en ambos nodos

```
ip addr | grep -A 2 ib0 | grep 'link/infiniband' | awk '{print $2}'
```

Comando usado para configurar el perfil de la conexión en el compute-1-28

```
nmcli con add type infiniband con-name ib0 transport-mode \
connected mtu 65520 infiniband.mac-address \
80:00:02:18:fe:80:00:00:00:00:00:00:00:02:c9:03:00:2f:c4:31 \
ipv4.method static ipv4.addresses 10.150.8.120 ipv4.gateway 10.150.8.120
```

Comando usado para configurar el perfil de la conexión en el compute-1-16

```
nmcli con add type infiniband con-name ib0 transport-mode \
connected mtu 65520 infiniband.mac-address \
80:00:02:18:fe:80:00:00:00:00:00:00:02:c9:03:00:2f:37:21 \
ipv4.method static ipv4.addresses 10.150.8.121 ipv4.gateway 10.150.8.120
```

Reunión 9 - Domingo 8 de Agosto 2021 (10 AM - 23:50 PM)

Asistentes

- Stiven Agudelo
- Miguel Correa
- Aurelio Vivas

Discusión

- Optimización del software y escritura de meta volante

Conclusiones y decisiones

- Nos dimos cuenta que se debería llegar al tamaño del problema más alto pero si era más alto el programa nunca se ejecutaba porque se dañaba la máquina; la memoria se llenaba.

https://software.intel.com/content/www/us/en/develop/documentation/onemkl-linux-developer-guide/top/intel-oneapi-math-kernel-library-benchmarks/intel-optimized-linpack-benchmark-for-linux.html

https://software.intel.com/content/www/us/en/develop/documentation/onemkl-linux-developer-guide/top/intel-oneapi-math-kernel-library-benchmarks/intel-distribution-for-linpack-benchmark/running-the-intel-distribution-for-linpack-benchmark.html

Experimento 1 (parámetros por defecto)

```
HPLinpack benchmark input file
Innovative Computing Laboratory, University of Tennessee
HPL.out      output file name (if any)
6            device out (6=stdout,7=stderr,file)
1            # of problems sizes (N)
1000         Ns
1            # of NBs
192 256      NBs
1            PMAP process mapping (0=Row-,1=Column-major)
1            # of process grids (P x Q)
1 2          Ps
1 2          Qs
16.0         threshold
1            # of panel fact
```

```
2 1 0        PFACTs (0=left, 1=Crout, 2=Right)
1            # of recursive stopping criterium
2            NBMINs (>= 1)
1            # of panels in recursion
2            NDIVs
1            # of recursive panel fact.
1 0 2        RFACTs (0=left, 1=Crout, 2=Right)
1            # of broadcast
0            BCASTs (0=1rg,1=1rM,2=2rg,3=2rM,4=Lng,5=LnM)
1            # of lookahead depth
0            DEPTHs (>=0)
0            SWAP (0=bin-exch,1=long,2=mix)
1            swapping threshold
1            L1 in (0=transposed,1=no-transposed) form
1            U  in (0=transposed,1=no-transposed) form
0            Equilibration (0=no,1=yes)
8            memory alignment in double (> 0)
```

```
 -- High Performance Computing Linpack Benchmark (HPL)
    HPL - 2.3 - December 2, 2018
    Antoine P. Petitet
    University of Tennessee, Knoxville
    Innovative Computing Laboratory
    (C) Copyright 2000-2008 All Rights Reserved
```

```
mpirun -genv I_MPI_FABRICS shm -n 1 --hosts 10.150.8.120
/opt/intel/oneapi/mkl/2021.3.0/benchmarks/mp_linpack/xhpl_intel64_dynamic
```

```
================================================================================
HPLinpack 2.3  --  High-Performance Linpack benchmark  --   December 2, 2018
Written by A. Petitet and R. Clint Whaley,  Innovative Computing Laboratory, UTK
Modified by Piotr Luszczek, Innovative Computing Laboratory, UTK
Modified by Julien Langou, University of Colorado Denver
================================================================================

An explanation of the input/output parameters follows:
T/V    : Wall time / encoded variant.
N      : The order of the coefficient matrix A.
NB     : The partitioning blocking factor.
P      : The number of process rows.
Q      : The number of process columns.
Time   : Time in seconds to solve the linear system.
Gflops : Rate of execution for solving the linear system.

The following parameter values will be used:

N        :    1000
NB       :     192
PMAP     : Column-major process mapping
P        :       1
Q        :       1
PFACT    :   Right
NBMIN    :       2
```

```
NDIV     :       2
RFACT    :   Crout
BCAST    :   1ring
DEPTH    :       0
SWAP     : Binary-exchange
L1       : no-transposed form
U        : no-transposed form
EQUIL    : no
ALIGN    :    8 double precision words

----------------------------------------------------------------------------

- The matrix A is randomly generated for each test.
- The following scaled residual check will be computed:
      ||Ax-b||_oo / ( eps * ( || x ||_oo * || A ||_oo + || b ||_oo ) * N )
- The relative machine precision (eps) is taken to be               1.110223e-16
- Computational tests pass if scaled residuals are less than                16.0

compute-1-28    : Column=000192 Fraction=0.005 Kernel=    0.00 Mflops=78725.20
compute-1-28    : Column=000384 Fraction=0.195 Kernel=54191.73 Mflops=67082.17
compute-1-28    : Column=000576 Fraction=0.385 Kernel=33097.20 Mflops=57086.77
compute-1-28    : Column=000768 Fraction=0.595 Kernel=15262.82 Mflops=48507.44
compute-1-28    : Column=000960 Fraction=0.795 Kernel= 4251.34 Mflops=42952.31
============================================================================
T/V                N    NB     P     Q               Time                 Gflops
----------------------------------------------------------------------------
WC00C2R2        1000   192     1     1               0.02            4.09680e+01
HPL_pdgesv() start time Sun Aug  8 11:48:12 2021

HPL_pdgesv() end time   Sun Aug  8 11:48:12 2021

----------------------------------------------------------------------------
||Ax-b||_oo/(eps*(||A||_oo*||x||_oo+||b||_oo)*N)=   6.78420035e-03 ...... PASSED
============================================================================

Finished      1 tests with the following results:
              1 tests completed and passed residual checks,
              0 tests completed and failed residual checks,
              0 tests skipped because of illegal input values.
----------------------------------------------------------------------------

End of Tests.
============================================================================
```

https://www.netlib.org/benchmark/hpl/tuning.html
https://www.cyberciti.biz/faq/how-to-install-htop-on-rhel-8-using-yum/

- NB, Según la documentación de netlib se recomienda que este valor sea entre 32 y 256. Nosotros escogimos 128. https://www.netlib.org/benchmark/hpl/faqs.html

1. Vamos a modificar Ns, Ps, Qs, NBs.

https://software.intel.com/content/www/us/en/develop/documentation/onemkl-linux-developer-guide/top/intel-oneapi-math-kernel-library-benchmarks/intel-distribution-for-linpack-benchmark/overview-of-the-intel-distribution-for-linpack-benchmark.html

```
[mpiuser@compute-1-16 mp_linpack]$ mpirun -np 1 --hosts 10.150.8.121
/opt/intel/oneapi/mkl/2021.3.0/benchmarks/mp_linpack/xhpl_intel64_dynamic -p 1 -q 1 -n 1000
================================================================================
HPLinpack 2.3  --  High-Performance Linpack benchmark  --   December 2, 2018
Written by A. Petitet and R. Clint Whaley,  Innovative Computing Laboratory, UTK
Modified by Piotr Luszczek, Innovative Computing Laboratory, UTK
Modified by Julien Langou, University of Colorado Denver
================================================================================

An explanation of the input/output parameters follows:
T/V    : Wall time / encoded variant.
N      : The order of the coefficient matrix A.
NB     : The partitioning blocking factor.
P      : The number of process rows.
Q      : The number of process columns.
Time   : Time in seconds to solve the linear system.
Gflops : Rate of execution for solving the linear system.

The following parameter values will be used:

N        :    1000
NB       :     128
PMAP     : Column-major process mapping
P        :       1
Q        :       1
PFACT    :   Right
NBMIN    :       2
NDIV     :       2
RFACT    :   Crout
BCAST    :   1ring
DEPTH    :       0
SWAP     : Binary-exchange
L1       : no-transposed form
U        : no-transposed form
EQUIL    : no
ALIGN    :    8 double precision words

--------------------------------------------------------------------------------

- The matrix A is randomly generated for each test.
- The following scaled residual check will be computed:
      ||Ax-b||_oo / ( eps * ( || x ||_oo * || A ||_oo + || b ||_oo ) * N )
- The relative machine precision (eps) is taken to be               1.110223e-16
- Computational tests pass if scaled residuals are less than                16.0

compute-1-16    : Column=000128 Fraction=0.005 Kernel=    0.00 Mflops=95418.97
compute-1-16    : Column=000256 Fraction=0.130 Kernel=73944.57 Mflops=84889.04
compute-1-16    : Column=000384 Fraction=0.260 Kernel=55951.04 Mflops=75780.03
compute-1-16    : Column=000512 Fraction=0.385 Kernel=38334.24 Mflops=67067.71
compute-1-16    : Column=000640 Fraction=0.515 Kernel=25451.33 Mflops=59919.15
compute-1-16    : Column=000768 Fraction=0.655 Kernel=14077.35 Mflops=53851.43
compute-1-16    : Column=000896 Fraction=0.795 Kernel= 5458.05 Mflops=48917.76
================================================================================
T/V                N    NB     P     Q               Time                 Gflops
--------------------------------------------------------------------------------
WC00C2R2        1000   128     1     1               0.02            4.42469e+01
```

```
HPL_pdgesv() start time Sun Aug  8 14:29:51 2021

HPL_pdgesv() end time   Sun Aug  8 14:29:51 2021

--------------------------------------------------------------------------------
||Ax-b||_oo/(eps*(||A||_oo*||x||_oo+||b||_oo)*N)=   6.22117400e-03 ...... PASSED
================================================================================

Finished      1 tests with the following results:
              1 tests completed and passed residual checks,
              0 tests completed and failed residual checks,
              0 tests skipped because of illegal input values.
--------------------------------------------------------------------------------

End of Tests.
================================================================================
```

```
[mpiuser@compute-1-16 mp_linpack]$ mpirun --hosts 10.150.8.121
/opt/intel/oneapi/mkl/2021.3.0/benchmarks/mp_linpack/xhpl_intel64_dynamic -p 1 -q 1 -n 1000
================================================================================
HPLinpack 2.3  --  High-Performance Linpack benchmark  --   December 2, 2018
Written by A. Petitet and R. Clint Whaley,  Innovative Computing Laboratory, UTK
Modified by Piotr Luszczek, Innovative Computing Laboratory, UTK
Modified by Julien Langou, University of Colorado Denver
================================================================================

An explanation of the input/output parameters follows:
T/V    : Wall time / encoded variant.
N      : The order of the coefficient matrix A.
NB     : The partitioning blocking factor.
P      : The number of process rows.
Q      : The number of process columns.
Time   : Time in seconds to solve the linear system.
Gflops : Rate of execution for solving the linear system.

The following parameter values will be used:

N        :    1000
NB       :     128
PMAP     : Column-major process mapping
P        :       1
Q        :       1
PFACT    :   Right
NBMIN    :       2
NDIV     :       2
RFACT    :   Crout
BCAST    :   1ring
DEPTH    :       0
SWAP     : Binary-exchange
L1       : no-transposed form
U        : no-transposed form
EQUIL    : no
ALIGN    :    8 double precision words

--------------------------------------------------------------------------------

- The matrix A is randomly generated for each test.
- The following scaled residual check will be computed:
      ||Ax-b||_oo / ( eps * ( || x ||_oo * || A ||_oo + || b ||_oo ) * N )
- The relative machine precision (eps) is taken to be               1.110223e-16
- Computational tests pass if scaled residuals are less than                16.0

compute-1-16    : Column=000128 Fraction=0.005 Kernel=    0.00 Mflops=  463.66
compute-1-16    : Column=000256 Fraction=0.130 Kernel=  369.73 Mflops=  418.27
compute-1-16    : Column=000384 Fraction=0.260 Kernel=  271.52 Mflops=  371.60
compute-1-16    : Column=000512 Fraction=0.385 Kernel=  190.65 Mflops=  329.95
```

```
compute-1-16    : Column=000640 Fraction=0.515 Kernel=  117.17 Mflops=  291.35
compute-1-16    : Column=000768 Fraction=0.655 Kernel=  160.16 Mflops=  283.32
compute-1-16    : Column=000896 Fraction=0.795 Kernel=  176.83 Mflops=  281.39
================================================================================
T/V                N    NB     P     Q               Time                 Gflops
--------------------------------------------------------------------------------
WC00C2R2        1000   128     1     1               2.43           2.75114e-01
HPL_pdgesv() start time Sun Aug  8 14:27:53 2021

HPL_pdgesv() end time   Sun Aug  8 14:27:55 2021

--------------------------------------------------------------------------------
||Ax-b||_oo/(eps*(||A||_oo*||x||_oo+||b||_oo)*N)=   6.22117400e-03 ...... PASSED
================================================================================

Finished      1 tests with the following results:
              1 tests completed and passed residual checks,
              0 tests completed and failed residual checks,
              0 tests skipped because of illegal input values.
--------------------------------------------------------------------------------

End of Tests.
```

```
[mpiuser@compute-1-16 mp_linpack]$ mpirun -n 1 --hosts 10.150.8.121 -genv
OMP_NUM_THREADS=2 -genv I_MPI_PIN_DOMAIN=omp
/opt/intel/oneapi/mkl/2021.3.0/benchmarks/mp_linpack/
xhpl_intel64_dynamic -p 1 -q 1 -n 1000
================================================================================
HPLinpack 2.3  --  High-Performance Linpack benchmark  --   December 2, 2018
Written by A. Petitet and R. Clint Whaley,  Innovative Computing Laboratory, UTK
Modified by Piotr Luszczek, Innovative Computing Laboratory, UTK
Modified by Julien Langou, University of Colorado Denver
================================================================================

An explanation of the input/output parameters follows:
T/V    : Wall time / encoded variant.
N      : The order of the coefficient matrix A.
NB     : The partitioning blocking factor.
P      : The number of process rows.
Q      : The number of process columns.
Time   : Time in seconds to solve the linear system.
Gflops : Rate of execution for solving the linear system.

The following parameter values will be used:

N        :    1000
NB       :     128
PMAP     : Column-major process mapping
P        :       1
Q        :       1
PFACT    :   Right
NBMIN    :       2
NDIV     :       2
RFACT    :   Crout
BCAST    :   1ring
DEPTH    :       0
SWAP     : Binary-exchange
L1       : no-transposed form
U        : no-transposed form
EQUIL    : no
ALIGN    :    8 double precision words
```

```
--------------------------------------------------------------------------------

- The matrix A is randomly generated for each test.
- The following scaled residual check will be computed:
      ||Ax-b||_oo / ( eps * ( || x ||_oo * || A ||_oo + || b ||_oo ) * N )
- The relative machine precision (eps) is taken to be               1.110223e-16
- Computational tests pass if scaled residuals are less than                16.0

compute-1-16    : Column=000128 Fraction=0.005 Kernel=    0.00 Mflops=94699.66
compute-1-16    : Column=000256 Fraction=0.130 Kernel=75243.51 Mflops=85280.81
compute-1-16    : Column=000384 Fraction=0.260 Kernel=55372.43 Mflops=75769.31
compute-1-16    : Column=000512 Fraction=0.385 Kernel=38392.46 Mflops=67084.10
compute-1-16    : Column=000640 Fraction=0.515 Kernel=25325.39 Mflops=59880.12
compute-1-16    : Column=000768 Fraction=0.655 Kernel=14219.82 Mflops=53892.42
compute-1-16    : Column=000896 Fraction=0.795 Kernel= 6167.99 Mflops=49532.83
================================================================================
T/V                N    NB     P     Q               Time                 Gflops
--------------------------------------------------------------------------------
WC00C2R2        1000   128     1     1               0.01            4.51737e+01
HPL_pdgesv() start time Sun Aug  8 15:42:04 2021
```