Benchmarking :-

It is a process to discover what is the best standard of performance seen in a specific company, by particular competitor or by a completely different industry

Theoretical Peak Performance :-

- Peak Performance is what the system is theoretically able to deliver
- Performance of a supercomputer is measured in Floating-point per second (FLOPS)
- Peak theoretical performance of a node

- Node Performance in GFlops = (CPU speed in GHz) x (number of CPU cores) x (CPU Instruction per cycle) x (number of CPU per node)

- Peak theoretial performance of a system:
  Peak theoretical performance of a node x no. of nodes.

Rmax & ~~Rmax~~ ~~Rmin~~ Rpeak :-

- In high-performance computing, Rmax & Rpeak are scores used to rank supercomputer
  Rpeak = theoretical Performance of systems
  Rmax = Standard (Actual performance)

HPC Efficiency:-

- HPC Efficiency is a measure (percentage) of the actual performance of HPC system against.

$$\text{Efficiency} = \frac{\text{Actual Performance GFLOPS}}{\text{Theoretical peak performance GFlops}}$$

$$= \frac{R_{max}}{R_{peak}}$$

Need of Green 500:-

the "performance per watt" metric is derived as

PPW = Performance / power

$$\text{GFlops per watt} = \frac{R_{max} \text{ (in GFlops)}}{P \text{ } (R_{max}) \text{ (in watt)}}$$

What is HPL?

- HPL measure how fast a computer ~~slaves~~ solves a dense n by n system of linear equations

  $Ax = b$

  i.e. solves a (random) dense linear system in double precision (64 bit) arithemetic on distributed - memory computers.

- HPL rely on an efficient implementation of the Basic linear Algebra Subprograms (BLAS)

$$N = \text{Sqrt}((\text{memory size in Gbytes} \times 1024 \times 1024 \times 1024 \times \text{Number of Nodes})/8) \times 0.80$$

N = Problem Size

NBs = Block size

HPL installation

systemctl stop firewalld.service
systemctl disable firewalld.service
vi /etc/selinux/config
↳ SELINUX = disabled
yum install epel-release
wget https://download.open-mpi.org/release/open-mpi/v4.1/openmpi-4.1.5.tar.gz
tar -xvf openmpi-4.1.5.tar.gz
./configure --prefix=/opt/openmpi-4.1.5
make -J 7 — gather mpi libraries
make install — install files in given prefix folder
yum install atlas
yum install atlas-devel

Export path :-

export PATH=$PATH:/opt/openmpi-4.1.5/bin
vim ~/.bashrc
↳ add lines at end of script
export ~~path~~ PATH=$PATH:/opt/openmpi-4.1.5./bin
source ~/.bashrc

rpm -ql atlas → shows where atlas packages are

wget

wget https://netlib.org/benchmark/hpl/hpl-2.3.tar.gz

tar -xvf hpl-2.3.tar.gz

~~cp Make~~ cd hpl-2.3/setup/

cp Make.Linux_PII_CBLAS /root/hpl-2.3/

vim /root/hpl-2.3/Make.Linux_PII_CBLAS

↳ TOPdir = /root/hpl-2.3 — line 70

MPdir = /opt/openmpi-4.1.5 — line 84

MPlib = $(MPdir)/lib/libmpi.so — line 86

LAdir = /usr/lib64/atlas ~~line~~ — line 95

LAlib = $(LAdir)/libtatlas.so.3 $(LAdir)/libsatlas.so.3

CC = /usr/bin/gcc → line 169

LINKER = /usr/bin/gcc → line 176

make arch = Linux_PII_~~CLABS~~ CBLAS

vi bin/Linux_PII_CBLAS/HPL.dat

↳ line 5 = 1

line 6 = 52352 (N = Problem size 32 GB RAM)

line 7 = 1

line 8 = 128 (~~No of~~ blocks size)

$$N = \sqrt{\frac{32 \times 1024 \times 1024 \times 1024}{8}} \times 0.80 = 65536 \times 0.80 = 52428.8$$

$$= \frac{52428.8}{128 \text{ (block size)}} = 409 \times 128 = \boxed{52352}$$

Line 10 = 1
Line 11 = 2 P
Line 12 = 4 Q (P ≤ Q our PC is 8 core)

$2 \times 4 = 8$

P Q

#CD bin/Linux_PII_CBLAS

ldd xhpl ⟶ list path of all dependencies

Vi ~/.bashrc

↳ export LD_LIBRARY_PATH = /opt/openmpi-4.1.5.
/lib : $LD_LIBRARY_PATH

Add these lines at-end

mpirun --allow-run-as-root -np 8 ./xhpl HPL.dat

no. of core

source ~/.bashrc