ADVANCED USE OF SUPERCOMPUTERS

Imad Kissami

Imad.kissami@um6p.ma

2023/2024

General knowledge

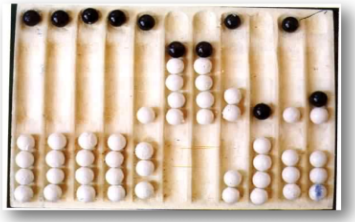
Definition of Computer

- Definition:
  - Computer is a programmable machine.
  - Computer is a machine that manipulates data according to a list of instructions.
  - Computer is any device which aids humans in performing various kinds of computations or calculations.
- Three principles characteristic of computer:
  - It responds to a specific set of instructions in a well defined manner.
  - It can execute a pre-recorded list of instructions.
  - It can quickly store and retrieve large amounts of data.

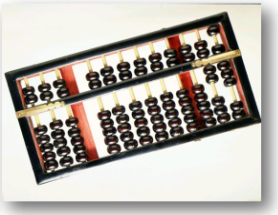
General knowledge

The Abacus

- The abacus, a simple counting aid, may have been invented in Babylonia (now Iraq) in the fourth century B.C.
- It used to perform basic arithmetic operations.



Earlier Abacus



Modern Abacus

General knowledge

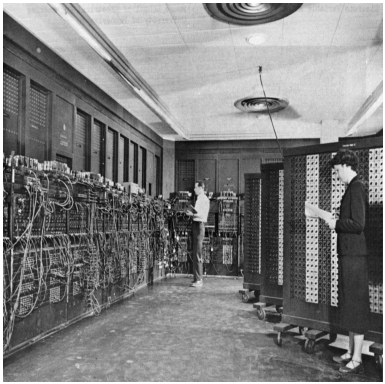
Jacquard Loom

- The Jacquard loom is a mechanical loom, invented by Joseph-Marie Jacquard in 1881.
- It an automatic loom controlled by punched cards.

General knowledge

The ENIAC

- ENIAC stands for Electronic Numerical Integrator and Computer.
- It was the first electronic general purpose computer.
- Completed in 1946.
- Developed by John Presper Eckert and John W. Mauchl.



General knowledge

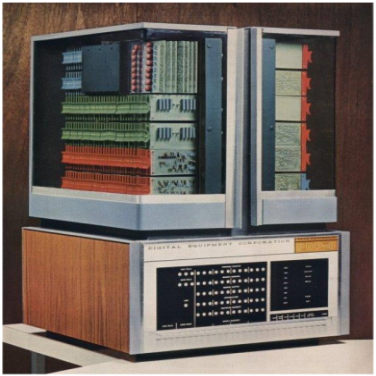
The IBM 360

- Developed by Gene Amdahl in 1965.
- It was the first family of computers designed to cover both commercial and scientific applications

General knowledge

The PDP-8

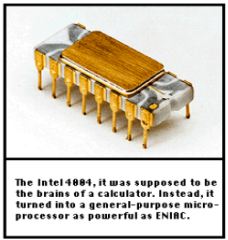
- Was introduced on 22 March 1965
- 12-bit minicomputer Produced by Digital Equipment Corporation (DEC).
- Priced at $18,500 (equivalent to about $150,000 in 2020)



General knowledge

The Microprocessor

- A computer chip that contains on it the entire CPU
  - Mass produced at a very low price
  - Computers become smaller and cheaper
- intel 4004 – the first computer on a chip, more powerful than the original ENIAC.
- Intel 8088 – used in IBM PC



General knowledge

Hardware

- Hardware – the physical devices that make up a computer (often referred to as the computer system)

General knowledge

Hardware core

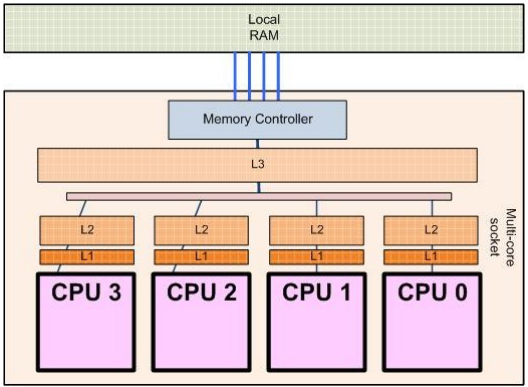
- CPU (Central Processing Unit)
  - CPU (machine) cycle – retrieve, decode, and execute instruction, then return result to RAM if necessary
  - CPU speed measured in gigahertz (GHz)
    - + GHz – number of billions of CPU cycles per seconds
- RAM (Random Access Memory)
  - Also called Memory, Main Memory, or Primary Storage
  - Measured in gigabytes (GB, billions of bytes) today
    - + Byte $->$ Character
  - RAM is volatile
    - + Temporary storage for instructions and data

General knowledge

Capacity of Secondary Storage Devices

- Kilobyte (KB or K) – about 1 thousand bytes
- Megabyte (MB or M or Meg) – about 1 million bytes
- Gigabyte (GB or Gig) – about 1 billion bytes
- Terabyte (TB) – about 1 trillion bytes

Modern architecture (CPU)



What's Supercomputer?

- Resource manager
  - A supercomputer is a computer with high level of performance compared to a general-purpose computer
  - Performance of a supercomputer is measured in floating point operations per second (FLOPS)

What is a supercomputer?

- cdc 6600: 1964 - three million calculations per second (3 MFLOPS)

What is a supercomputer?

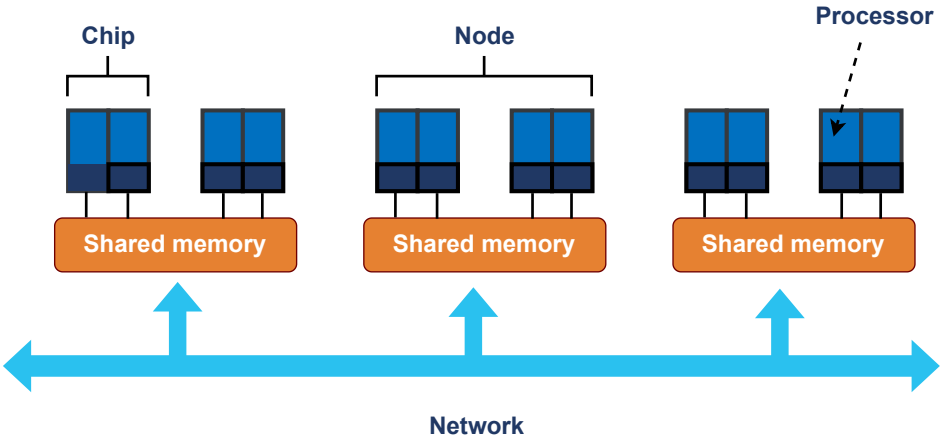
- Summit: 2018 - 36000 cores - 200 quadrillion calculations per second
- Toubkal: 2021 - 71,232 cores (5.01 PFLOPS)
- SimLab: 2014 - 696 cores

What is a supercomputer?

- Frontier: 2022 - 8 million cores - AMD EPYC with 64 cores and speed up to 2GHz - quintillion calculations per second (1,102 exaFLOPS)

What is a supercomputer?

Cluster



What's slurm?

- Resource manager
  - Manage pool of computational resources
  - Allocate resources based on constraints (node count, CPU count, available memory, GPUs)
  - Track active of allocated resources

What's slurm?

- Job scheduler
  - Accept job submissions (description of work to be performed, resource constraints)
  - Order jobs for execution (priority queue)
  - Execute jobs (allocate resources, configure environment, launch tasks)
  - Track resources usage

Submitting jobs

- A job script contains the command(s) to be executed when the job eventually executes
  - Example job scripts can be found at https://github.com/HPC-Simlab/Tutorials
  - Header at top of job script can include resource constraints and other job-specific information for slurm
    - The same flags you would enter on the command line, prefixed with #SBATCH

Submitting jobs

- CPU/memory resource constraints

Flag	Description	Example Use
--nnodes=# -N#	Job should span this many nodes; defaults to 1	
--ntasks-per-node=#	Each node gets this many tasks	MPI workers
--cpus-per-task=# -c#	Each task should have this many CPU cores associated with it; defaults to 1	OpenMP/threading; hybrid MPI
--mem=#	Allocate this much memory *per node* to the task(s) on that node; units of K/M/G/T are allowed	--mem=0 (all memory on node)
--mem-per-cpu=#	Allocate this much memory *per CPU* to the task(s) on that node; units of K/M/G/T are allowed	

Submitting jobs

- Time constraints

Flag	Description	Example Use
--begin=<time>	Job should not start until after this date/time	Delay for data to be ready for d/l
--deadline=<time>	Remove job is it cannot complete before this date/time	If results won't be ready by date, why bother running the job
--time=<time>	Job will run no longer than this duration (wall time limit); defaults to 30 minutes	Necessary on all jobs
--time-min=<time>	Provide a lower bound to the wall time limit, making --time an upper bound	Useful for backfill (flexible wall time makes for easier fit)

Submitting jobs

- Time constraints

Format	Description
HH:MM[:SS]	A time either today or (if already passed) tomorrow Also acts as a duration
D-HH:MM[:SS]	Duration including *D* 24-hour periods
YYYY-MM-DD[THH:MM[:SS]]	Specific date and time; midnight is implied sans [THH:MM[:SS]]
today tomorrow	Midnight this day or the next
midnight noon fika teatime	Specific time of day on this day; "fika" is Swedish coffee break (3 p.m.) and "teatime" is English tea (4 p.m.)
now+<offset>	Current date and time plus an offset (in min, hr, day, week)

Submitting jobs

- Partitions on SimLab:
  - **defq:** partition is automatically used if no partition is specified by all jobs.
  - **shortq:** partition used for short jobs (max. 12 hours)
  - **longq:** partition used for long jobs (max 30 days)
  - **special:** used for running parallel jobs (max 30 minutes).
  - **visu:** partition used for visualization.
  - **gpu:** partition used for gpu computations (all nodes in this partition have gpu card V100 or P40)

Submitting jobs

- Partitions on SimLab:

Partition	Max. Cpu Time	Nodes available for the partition	Max nodes per job	Min-Max cores per job
defq	4 hours	5 (node01, node02 node03, node14, node15)	1	1-44
shortq	12 hours	5 (node01, node02 node03, node14, node15)	2	1-88
longq	30 days	5 (node01, node02 node03, node14, node15)	1	1-44
special	30 minutes	15 (all nodes)	15	1-652
visu	24 hours	1 (visu01)	1	1-44
gpu	48 hours	12 (node[06-17])	2	1-88

Job status

- There are three unique ways to check job status with Slurm
  - Pending and running jobs:
    - squeue
    - scontrol
  - Completed jobs
    - sacct

Job status

- **squeue:** information about jobs in the scheduling queue
  defaut is to show all jobs in all states (except completed)

Flag	Description	Example Use
--partition=<name>	Limit to jobs using a specific partition(s)	--partition=devel,standard
--account=<workgroup>	Limit to jobs submitted by specific workgroup(s)	--account=it_nss,it_css
--user=<uname>	Limit to jobs submitted by specific user(s)	--user=frey,anita
--format=<output_format>	User-defined printf()-like format for each matched job	--format=%all
--sort=<fields>	Order matched jobs before displaying	--sort=P,t,-p
--jobs=<jobid>	Limit to one or more specific jobs by id	--jobs=4,8,15,16,23,42

Job status

- **scontrol:** view job state

```
[ikissami@frontend01 ~]$ scontrol show job 5835318
JobId=5835318 JobName=jupyter-notebook
   UserId=hamsbaai(1065) GroupId=hamsbaai(1076) MCS_label=N/A
   Priority=102841 Nice=0 Account=d4r-account QOS=normal
   JobState=RUNNING Reason=None Dependency=(null)
   Requeue=1 Restarts=0 BatchFlag=1 Reboot=0 ExitCode=0:0
   RunTime=00:20:07 TimeLimit=1-20:10:00 TimeMin=N/A
   SubmitTime=2023-09-05T14:10:06 EligibleTime=2023-09-05T14:10:06
   StartTime=2023-09-05T14:10:06 EndTime=2023-09-07T10:20:06 Deadline=N/A
   PreemptTime=None SuspendTime=None SecsPreSuspend=0
   LastSchedEval=2023-09-05T14:10:06
   Partition=gpu AllocNode:Sid=node11:294033
   ReqNodeList=(null) ExcNodeList=(null)
   NodeList=node16
   BatchHost=node16
   NumNodes=1 NumCPUs=44 NumTasks=1 CPUs/Task=1 ReqB:S:C:T=0:0:*:*
   TRES=cpu=44,mem=385308M,node=1,billing=44
   Socks/Node=* NtasksPerN:B:S:C=0:0:*:* CoreSpec=*
   MinCPUsNode=1 MinMemoryCPU=8757M MinTmpDiskNode=0
   Features=(null) DelayBoot=00:00:00
   Gres=gpu:1 Reservation=(null)
   OverSubscribe=NO Contiguous=0 Licenses=(null) Network=(null)
   Command=/home/hamsbaai/fcisJupyter.sh
   WorkDir=/home/hamsbaai
   StdErr=/home/hamsbaai/jupyter-notebook-%J.log
   StdIn=/dev/null
   StdOut=/home/hamsbaai/jupyter-notebook-%J.log
   Power=
```

Job status

- **scontrol:** view job state

```
[ikissami@frontend01 ~]$ scontrol show node node01
NodeName=node01 Arch=x86_64 CoresPerSocket=20
   CPUAlloc=32 CPUErr=0 CPUTot=40 CPULoad=32.22
   AvailableFeatures=broadwell
   ActiveFeatures=broadwell
   Gres=(null)
   NodeAddr=node01 NodeHostName=node01 Version=17.11
   OS=Linux 3.10.0-862.el7.x86_64 #1 SMP Wed Mar 21 18:14:51 EDT 2018
   RealMemory=128823 AllocMem=103040 FreeMem=111563 Sockets=2 Boards=1
   State=MIXED ThreadsPerCore=1 TmpDisk=51175 Weight=1 Owner=N/A MCS_label=N/A
   Partitions=defq,shortq,longq,special
   BootTime=2023-08-28T18:47:17 SlurmdStartTime=2023-08-28T18:49:19
   CfgTRES=cpu=40,mem=128823M,billing=40
   AllocTRES=cpu=32,mem=103040M
   CapWatts=n/a
   CurrentWatts=0 LowestJoules=0 ConsumedJoules=0
   ExtSensorsJoules=n/s ExtSensorsWatts=0 ExtSensorsTemp=n/s
```

Job status

- **sacct:** view data from job accounting log/database
  - Once job completes execution, the only way to access its info
  - includes:
    - exit code and state
    - total resource utilization

Flag	Description	Example Use
--starttime=<date-time>	Limit to jobs in scheduling queue after given date/time; default is midnight of current day	--starttime=2019-09-01
--endtime=<date-time>	Limit to jobs in scheduling queue before given date/time; default is the current date and time	--endtime=2019-10-01
--name=<name-list>	Limit to jobs with the given name(s)	--name=hello,world
--state=<state_list>	Limit to jobs in the given state(s)	--state=OOM,PR
--format=<fields>	List of fields to display (with optional widths)	--format=ALL