Processamento Paralelo - II

Canto III

A porta do Inferno - Vestíbulo
Rio Aqueronte - Caronte

POR MIM SE VAI À CIDADE DOLENTE,
POR MIM SE VAI À ETERNA DOR ,
POR MIM SE VAI À PERDIDA GENTE.

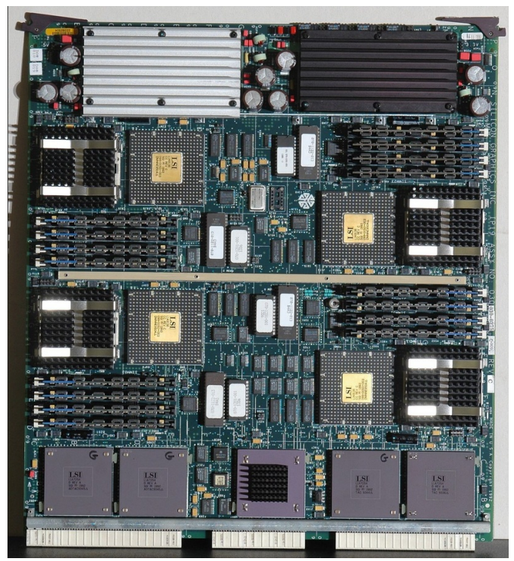
JUSTIÇA MOVEU O MEU ALTO CRIADOR,
QUE ME FEZ COM O DIVINO PODER,
O SABER SUPREMO E O PRIMEIRO AMOR.

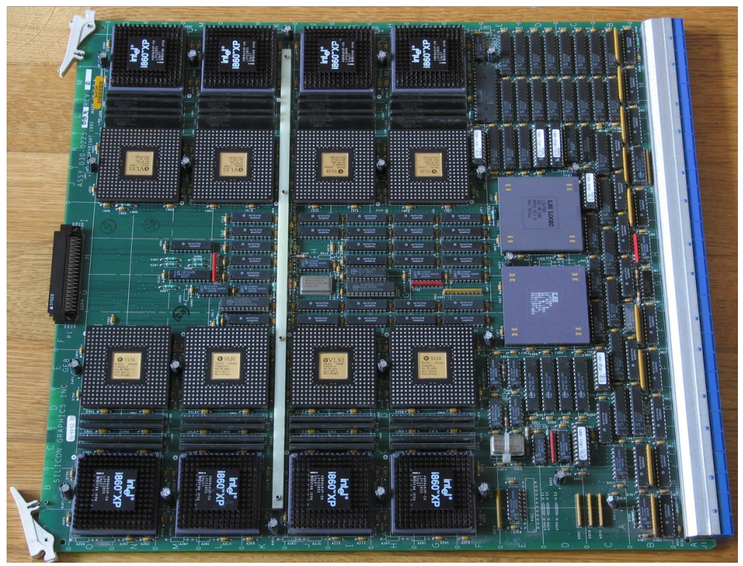
ANTES DE MIM COISA ALGUMA FOI CRIADA
EXCETO COISAS ETERNAS, E ETERNA EU
DURO.
DEIXAI TODA ESPERANÇA, VÓS QUE ENTRAIS!

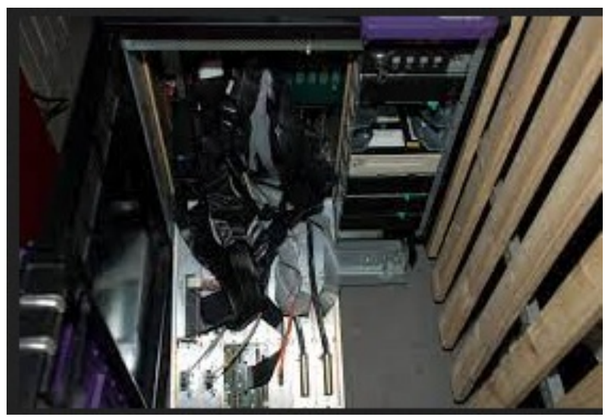
https://www.stelle.com.br/pt/inferno/canto_3.html

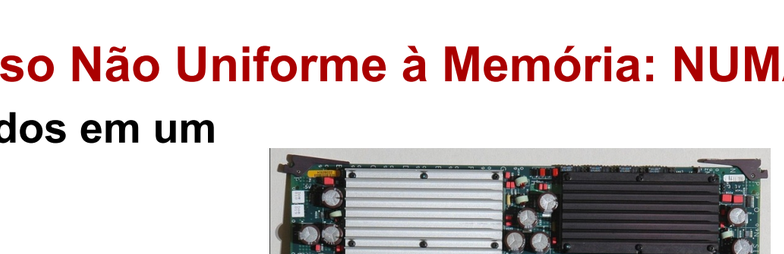
Acesso Não Uniforme à Memória: NUMA

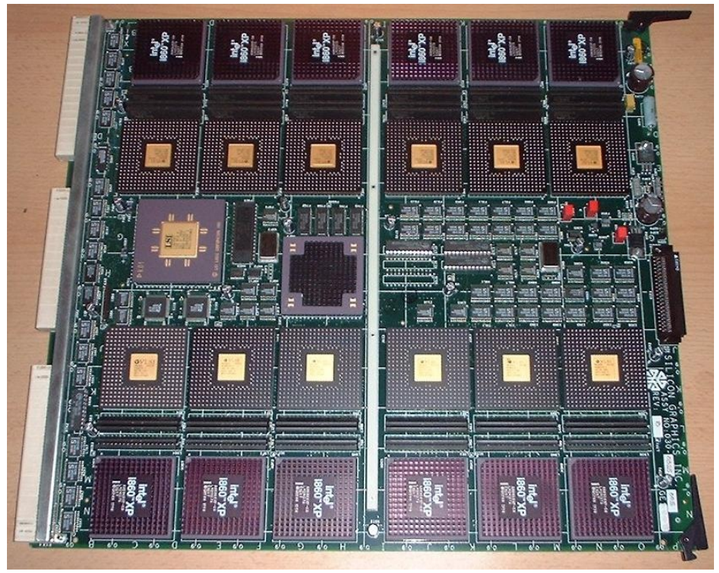
Quantos processadores podem ser colocados em um mesmo computador?











Acesso Não Uniforme à Memória: NUMA

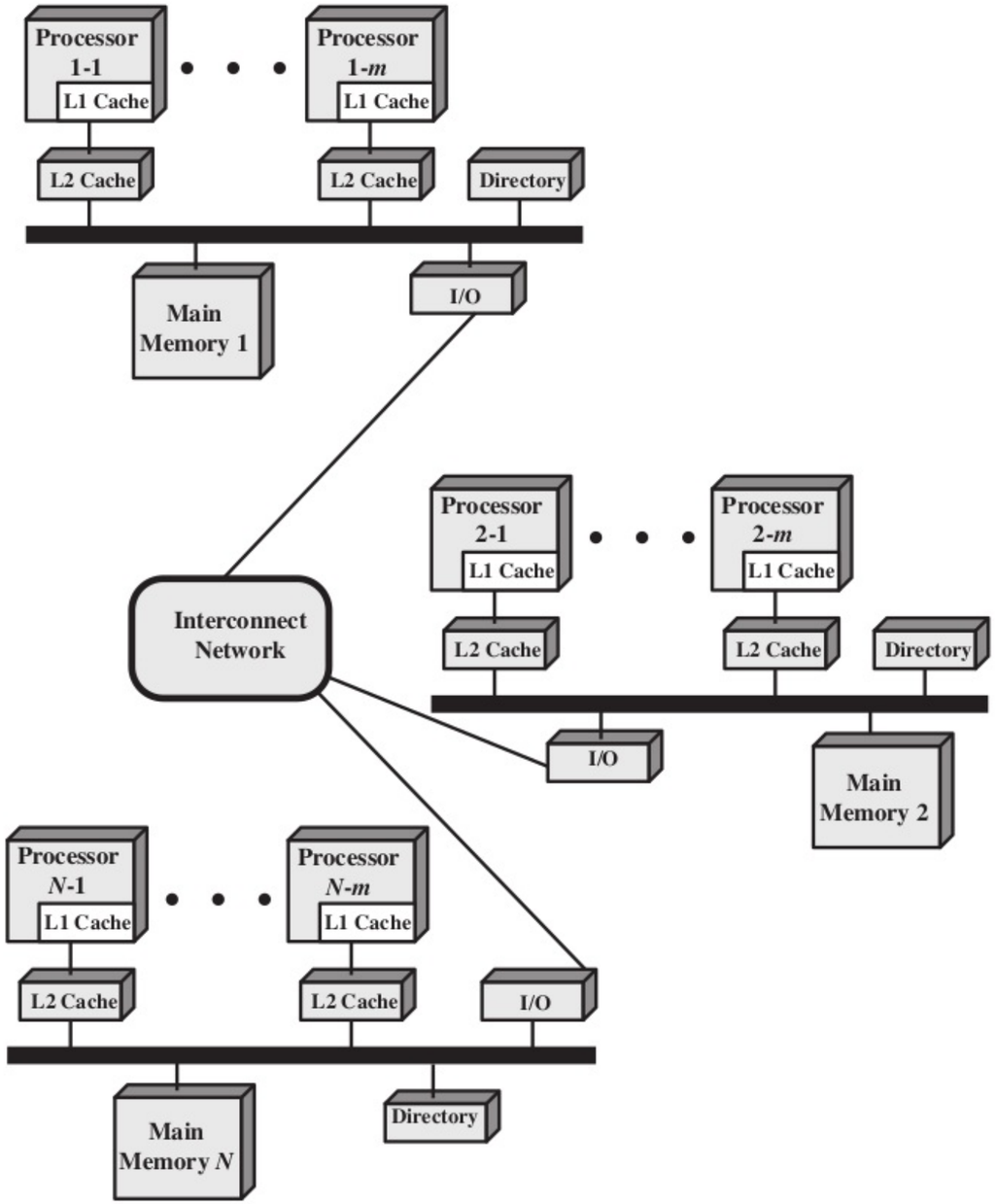


Figure 17.11 CC-NUMA Organization

Acesso Uniforme à Memória:
- Uniform Memory Access (UMA)
- Todos os processadores têm acesso a toda memória
- O tempo de acesso à memória é igual para todos os endereços
- O tempo de acesso é igual para diferentes processadores

Acesso Não Uniforme à Memória:
- Non-Uniform Memorory Access (NUMA)
- O tempo de cesso à memória difere dependendo do endereço da memória a ser acessado
- O tempo de acesso é diferente para processadores diferentes

Acesso Não Uniforme à Memória com Coerência de Cache:
- NUMA com coerência de cache (NUMA-CC)

Como NUMA-CC funciona?

Atenção: não confunda NUMA com Cluster!

Clusters

Cluster: grupo de computadores completos trabalhando juntos como um recurso computacional unificado que cria a ilusão de ser uma única máquina.

Vantagens:

- **Escalabilidade absoluta**
- **Escalabilidade incremental**
- **Alta disponibilidade**
- **Custo/benefício**

Clusters Beowulf

https://spinoff.nasa.gov/Spinoff2020/it_1.html
https://ntrs.nasa.gov/citations/20150001285

Thomas Sterling
Donald Becker

Hardware:
comum ou "comum"

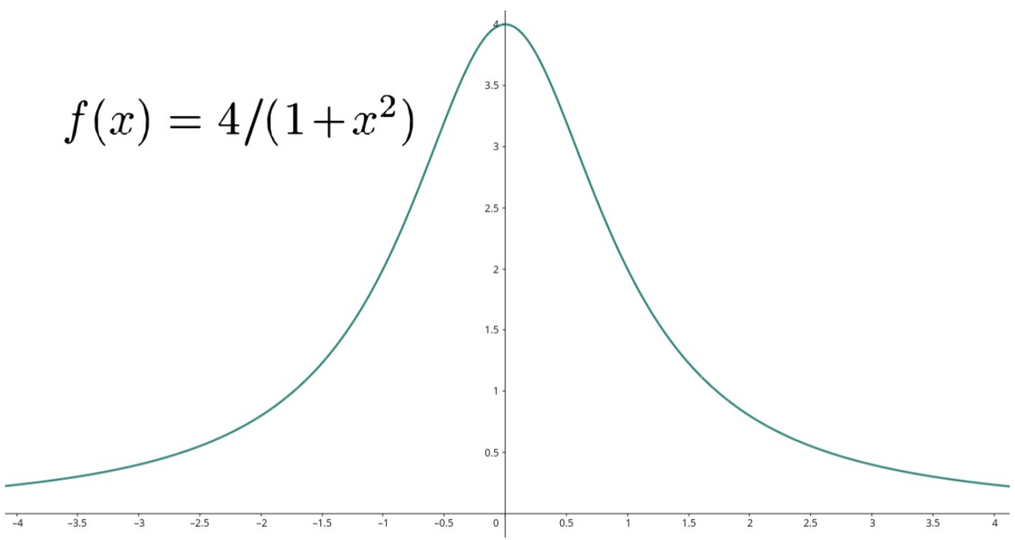
Sistema operacional:
Linux ou Unix-Like

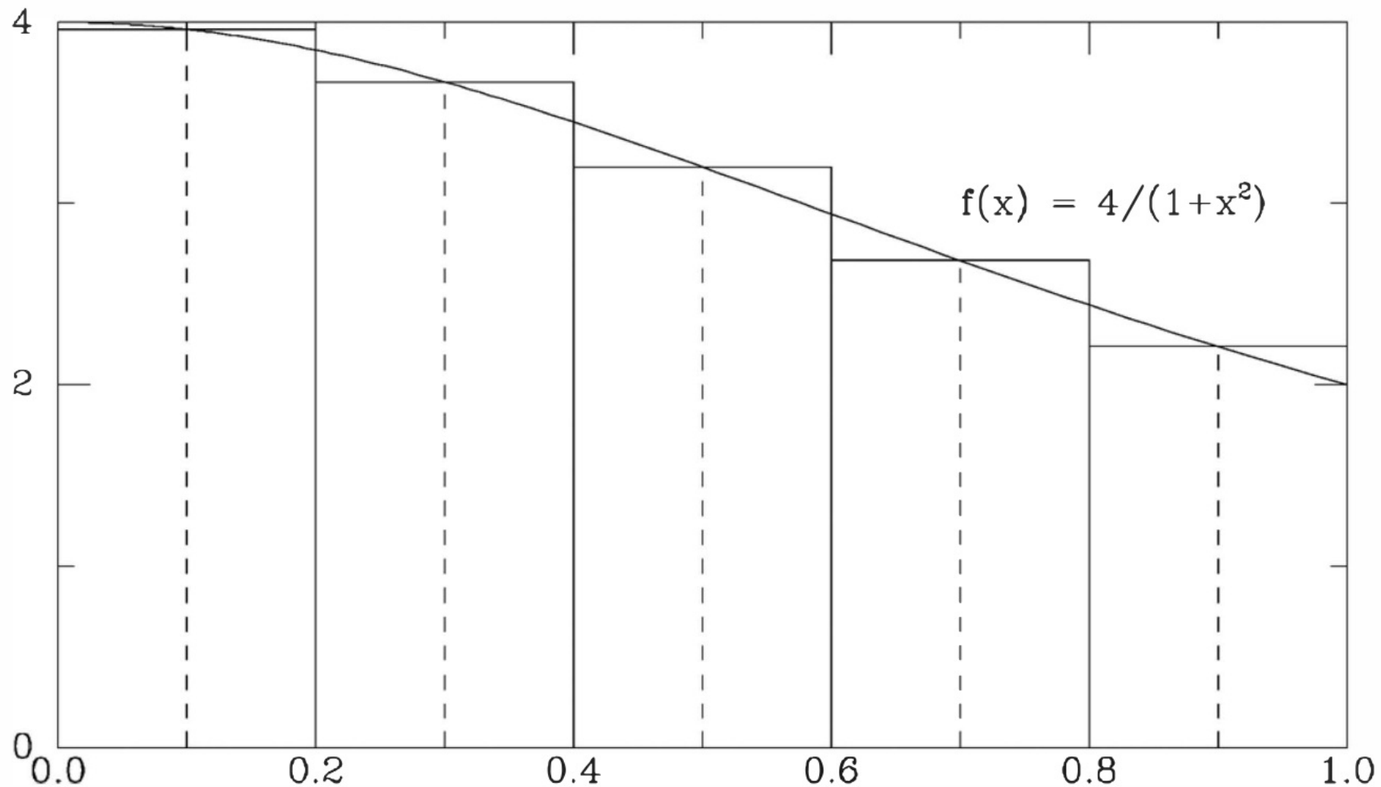
Bibliotecas para proc. paralelo:
Message Passing Interface (MPI)
Parallel Virtual Machine (PVM)

- Exemplos:
Open MPI
MPICH

Modo Geral de Computação em Cluster

$$\int_0^1 \frac{1}{1+x^2} dx = \arctan(x)|_0^1 = \arctan(1) - \arctan(0) = \arctan(1) = \frac{\pi}{4}$$





Supercomputadores no Top 500: clusters

Top 10 positions of the 58th TOP500 in November 2021[34]

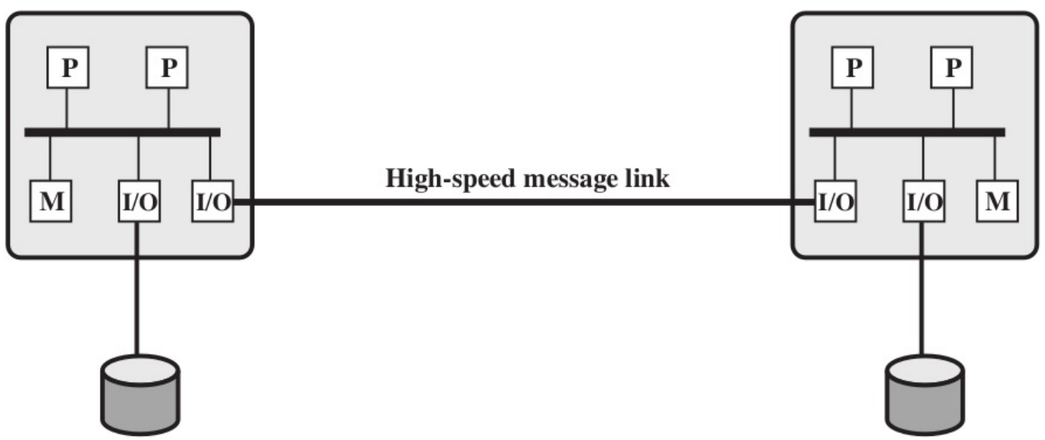
Rank (previous)	Rmax Rpeak (PFLOPS)	Name	Model	CPU cores	Accelerator (e.g. GPU) cores	Interconnect	Manufacturer	Site country	Year	Operating system
1 — (1)	442.010 537.212	Fugaku	Supercomputer Fugaku	158,976 × 48-core Fujitsu A64FX @2.2 GHz	0	Tofu interconnect D	Fujitsu	RIKEN Center for Computational Science Japan	2020	Linux (RHEL)
2 — (2)	148.600 200.795	Summit	IBM Power System AC922	9,216 × 22-core IBM POWER9 @3.07 GHz	27,648 × 80 Nvidia Tesla V100	InfiniBand EDR	IBM	Oak Ridge National Laboratory United States	2018	Linux (RHEL 7.4)
3 — (3)	94.640 125.712	Sierra	IBM Power System S922LC	8,640 × 22-core IBM POWER9 @3.1 GHz	17,280 × 80 Nvidia Tesla V100	InfiniBand EDR	IBM	Lawrence Livermore National Laboratory United States	2018	Linux (RHEL)
4 — (4)	93.015 125.436	Sunway TaihuLight	Sunway MPP	40,960 × 260-core Sunway SW26010 @1.45 GHz	0	Sunway[35]	NRCPC	National Supercomputing Center in Wuxi China[35]	2016	Linux (RaiseOS 2.0.5)
5 — (5)	64.590 89.795	Perlmutter	HP	? × ?-core AMD Epyc 7763 64-core @2.45 GHz	? × 108 Nvidia Ampere A100	Slingshot-10	HPE	NERSC United States	2021	Linux (HPE Cray OS)
6 — (6)	63.460 79.215	Selene	Nvidia	1,120 × 64-core AMD Epyc 7742 @2.25 GHz	4,480 × 108 Nvidia Ampere A100	Mellanox HDR Infiniband	Nvidia	Nvidia United States	2020	Linux (Ubuntu 20.04.1)
7 — (7)	61.445 100.679	Tianhe-2A	TH-IVB-FEP	35,584 × 12-core Intel Xeon E5–2692 v2 @2.2 GHz	35,584 × Matrix-2000[36] 128-core	TH Express-2	NUDT	National Supercomputer Center in Guangzhou China	2013	Linux (Kylin)
8 — (8)	44.120 70.980	JUWELS (booster module) [37][38]	BullSequana XH2000	1,872 × 24-core AMD Epyc 7402 @2.8 GHz	3,744 × 108 Nvidia Ampere A100	Mellanox HDR Infiniband	Atos	Forschungszentrum Jülich Germany	2020	Linux (CentOS)
9 — (9)	35.450 51.721	HPC5	Dell	3,640 × ?-core Intel Xeon Gold 6252 @2.1 GHz	7,280 × 80 Nvidia Tesla V100	Mellanox HDR Infiniband	Dell EMC	Eni Italy	2020	Linux (CentOS 7)
10 NEW	30.050 39.531	Voyager-EUS2	ND96AMSR_A100_V4	5,280 × 48-core AMD Epyc 7V12 @2.45 GHz	? × Nvidia A100	Mellanox HDR Infiniband	Microsoft Azure	Azure East US 2 United States	2021	Linux (Ubuntu 18.04)

https://www.top500.org

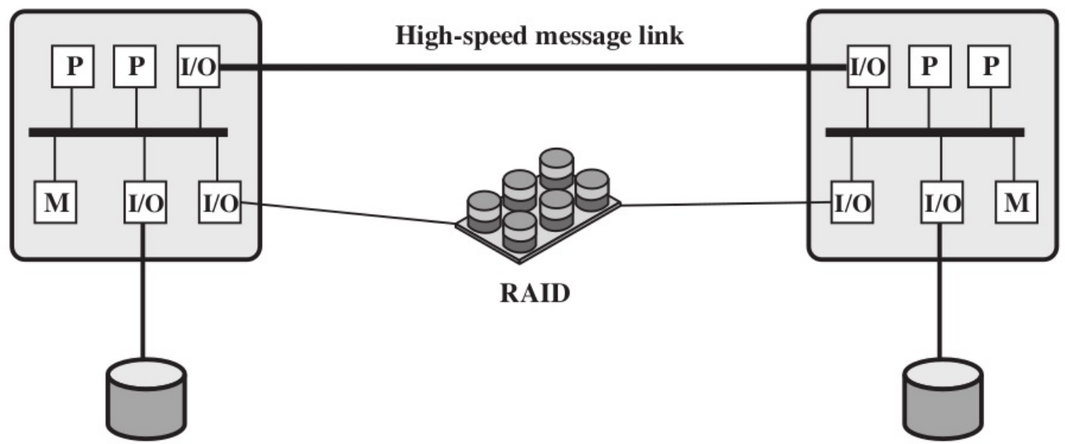
Clusters: configurações

Clustering Method	Description	Benefits	Limitations
Passive Standby	A secondary server takes over in case of primary server failure.	Easy to implement.	High cost because the secondary server is unavailable for other processing tasks.
Active Secondary:	The secondary server is also used for processing tasks.	Reduced cost because secondary servers can be used for processing.	Increased complexity.
Separate Servers	Separate servers have their own disks. Data is continuously copied from primary to secondary server.	High availability.	High network and server overhead due to copying operations.
Servers Connected to Disks	Servers are cabled to the same disks, but each server owns its disks. If one server fails, its disks are taken over by the other server.	Reduced network and server overhead due to elimination of copying operations.	Usually requires disk mirroring or RAID technology to compensate for risk of disk failure.
Servers Share Disks	Multiple servers simultaneously share access to disks.	Low network and server overhead. Reduced risk of downtime caused by disk failure.	Requires lock manager software. Usually used with disk mirroring or RAID technology.

Clusters: configurações



(a) Standby server with no shared disk



(b) Shared disk

Clusters: questões importantes

Gerenciamento de falhas:

- clusters de ALTA DISPONIBILIDADE
- clusters de ALTA TOLERÂNCIA A FALHAS

Recuperação de falhas:

- FAILOVER:
  - é a troca de aplicações, recursos e dados de um nó que falhou para um outro nó no cluster
- FAILBACK:
  - é a restauração de aplicações, recursos e dados para o nó original, após o mesmo ser consertado

Computação paralela:

- exige bibliotecas e softwares específicos

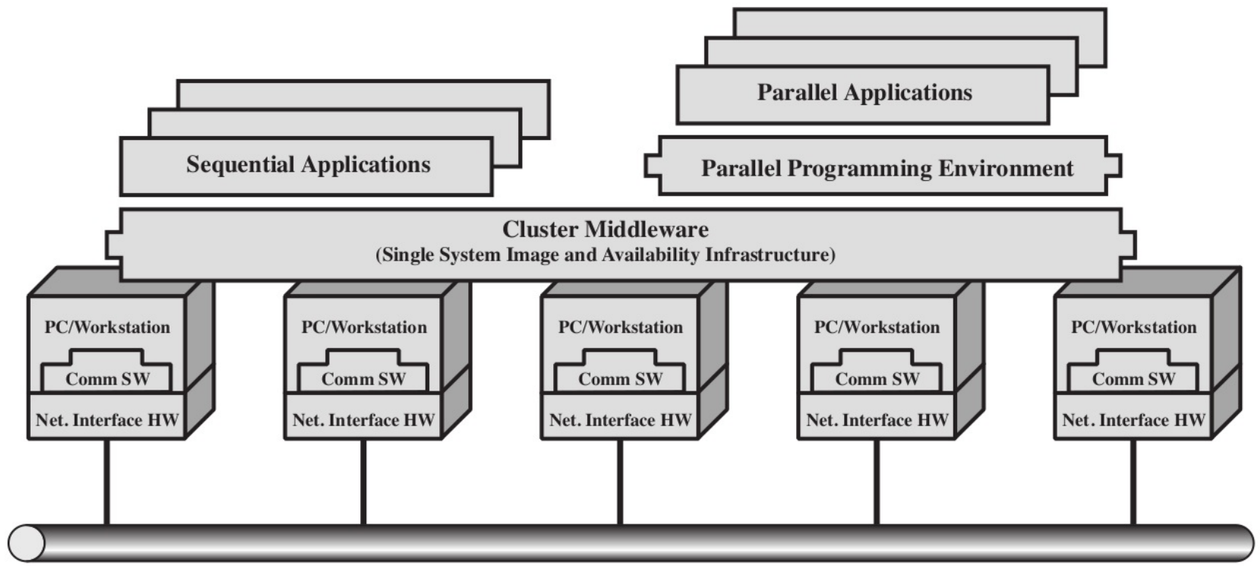
```c
#include <stdio.h>
int main(void) {
        printf("hello, world\n");
        return 0;
}
```

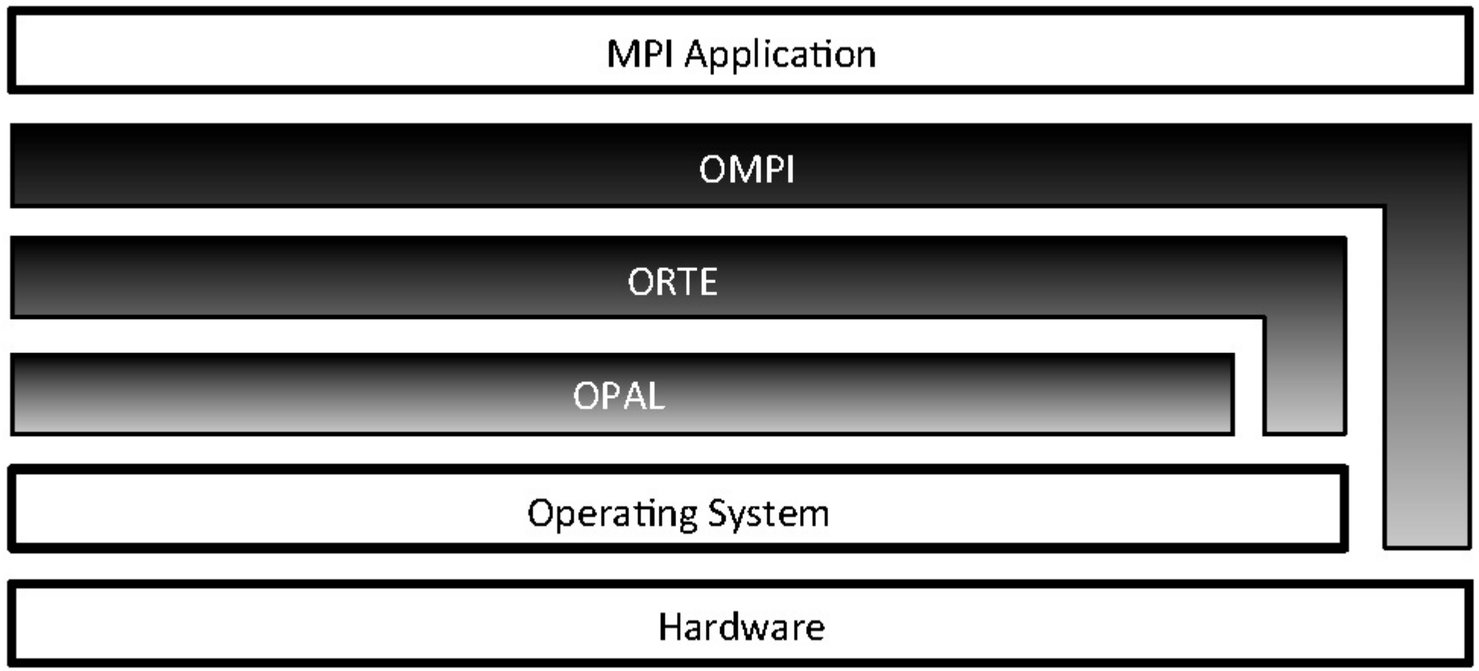
```c
#include <mpi.h>
#include <stdio.h>

int main(int argc, char** argv) {
    // Initialize the MPI environment
    MPI_Init(NULL, NULL);

    // Get the number of processes
    int world_size;
    MPI_Comm_size(MPI_COMM_WORLD, &world_size);

    // Get the rank of the process
    int world_rank;
    MPI_Comm_rank(MPI_COMM_WORLD, &world_rank);

    // Get the name of the processor
    char processor_name[MPI_MAX_PROCESSOR_NAME];
    int name_len;
    MPI_Get_processor_name(processor_name, &name_len);

    // Print off a hello world message
    printf("Hello world from processor %s, rank %d out of %d processors\n",
           processor_name, world_rank, world_size);

    // Finalize the MPI environment.
    MPI_Finalize();
}
```

```
Compiling
Compilation is OK
Execution ...
Hello world from processor 6f1ebfdf9e88, rank 0 out of 4 processors
Hello world from processor 6f1ebfdf9e88, rank 1 out of 4 processors
Hello world from processor 6f1ebfdf9e88, rank 2 out of 4 processors
Hello world from processor 6f1ebfdf9e88, rank 3 out of 4 processors
```

https://www.codingame.com/playgrounds/47058/have-fun-with-mpi-in-c/lets-start-to-have-fun-with-mpi

Clusters: visão de máquina única





Clusters: beowulf x blade

Clusters: Manual do Usuário de um Supercomputador

Frontera User Guide

Last update: December 6, 2021 Download PDF

STATUS UPDATES AND NOTICES

- **All users: refer to updated Remote Desktop Access instructions.** (07/21/2021)
- **New Queue: A new queue: " `small` " has been created specifically for one and two node jobs.** Jobs of one or two nodes that will run for up to 48 hours should be submitted to this new `small` queue. The `normal` queue now has a lower limit of three nodes for all jobs. These new limits will improve the turnaround time for all jobs in the `normal` and `small` queues. (03-30-21)
- **Frontera has new large memory nodes**, accessible via the `nvdimm` queue. (04/03/20)
- Users now have access to additional Frontera User Portal functionality. **Log in the portal to access your dashboard and other features.** (03-25-20)
- TACC Staff have put forth new file system and job submission guidelines. **All users: read Managing I/O on TACC Resources.** (01/09/20)
- **Frontera's GPU queues, `rtx` and `rtx-dev` are now open.** See Frontera's production queues for more information. Execute `qlimits` to display Frontera's queue configurations and charge rates. (12/07/19)
- **The TACC Visualization Portal now supports Frontera job submission for VNC and DCV remote desktops as well as Jupyter Notebook sessions.** We recommend submitting to the `development queue` for fastest turn-around. (12/05/2019)
- **Frontera's `flex` queue is now open.** Jobs in this queue are charged a lower rate (.8 SUs) but are pre-emptable after a guaranteed runtime of at least one hour. (10/30/2019)
- Please run all your jobs out of the `$SCRATCH` filesystem, instead of `$WORK`, to preserve the stability of the system. (10/10/2019)
- You may now subscribe to Frontera User News. Stay up-to-date on Frontera's status, scheduled maintenances and other notifications. (10/10/2019)
- **All users: read the Good Citizenship section.** Frontera is a shared resource and your actions can impact other users. (10/10/2019)

https://frontera-portal.tacc.utexas.edu/user-guide/

Referência e Leitura Adicional

Capítulo 17: Processamento Paralelo

- **17.6 Acesso não uniforme à memória**

- **17.5 Clusters**