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

Multithreading e CPUs Multicores

Desempenho de processadores ao longo do tempo (MIPS):

Processor / System	MIPS	Year
UNIVAC I	0.002 MIPS at 2.25 MHz	1951
IBM 7030 ("Stretch")	1.200 MIPS at 3.30 MHz	1961
CDC 6600	10.00 MIPS at 10.00 MHz	1965
Intel 4004	0.092 MIPS at 0.740 MHz	
IBM System/370 Model 158	0.640 MIPS at 8.696 MHz	1972
Intel 8080	0.290 MIPS at 2.000 MHz	
Cray 1	160.0 MIPS at 80.00 MHz	1975
MOS Technology 6502	0.430 MIPS at 1.000 MHz	1975
Intel 8080A	0.435 MIPS at 3.000 MHz	
Zilog Z80	0.580 MIPS at 4.000 MHz	
Motorola 6802	0.500 MIPS at 1.000 MHz	1977
IBM System/370 Model 158-3	0.730 MIPS at 8.696 MHz	1977
VAX-11/780	1.000 MIPS at 5.000 MHz	1977
Motorola 6809	0.420 MIPS at 1.000 MHz	1978
Intel 8086	0.330 MIPS at 5.000 MHz	1978
Fujitsu MB8843	2.000 MIPS at 2.000 MHz	
Intel 8088	0.750 MIPS at 10.00 MHz	1979
Motorola 68000	1.400 MIPS at 8.000 MHz	1979
Zilog Z8001/Z8002	1.5 MIPS at 6 MHz	1979
Intel 8035/8039/8048	6 MIPS at 6 MHz	
Fujitsu MB8843/MB8844	6 MIPS at 6 MHz	
Zilog Z80/Z80H	1.16 MIPS at 8 MHz	

Multithreading e CPUs Multicores

Processor / System	MIPS	Year
Motorola 6802	1.79 MIPS at 3.58 MHz	1981
Zilog Z8001/Z8002B	2.5 MIPS at 10 MHz	1981
MOS Technology 6502	2.522 MIPS at 5.865 MHz	1981
Intel 80286	1.28 MIPS at 12 MHz	1982
Motorola 68000	2.188 MIPS at 12.5 MHz	1982
Motorola 68010	2.407 MIPS at 12.5 MHz	1982
NEC V20	4 MIPS at 8 MHz	
LINKS-1 Computer Graphics System (257-processor)	642.5 MIPS at 10 MHz	1982
Texas Instruments TMS32010	5 MIPS at 20 MHz	1983
NEC V30	5 MIPS at 10 MHz	
Motorola 68010	3.209 MIPS at 16.67 MHz	1984
Motorola 68020	4.848 MIPS at 16 MHz	1984
Hitachi HD63705	2 MIPS at 2 MHz	1985
Intel i386DX	2.15 MIPS at 16 MHz	1985
Hitachi-Motorola 68HC000	3.5 MIPS at 20 MHz	1985
Intel 8751	1 MIPS at 12 MHz	1985
Sega System 16 (4-processor)	16.33 MIPS at 10 MHz	1985
ARM2	4 MIPS at 8 MHz	1986
Texas Instruments TMS34010	6 MIPS at 50 MHz	1986
NEC V70	6.6 MIPS at 20 MHz	1987
Motorola 68030	9 MIPS at 25 MHz	1987
Gmicro/200	10 MIPS at 20 MHz	1987
Texas Instruments TMS320C20	12.5 MIPS at 25 MHz	1987
Analog Devices ADSP-2100	12.5 MIPS at 12.5 MHz	1987
Texas Instruments TMS320C25	25 MIPS at 50 MHz	1987
Motorola 68020	10 MIPS at 33 MHz	1988
Motorola 68030	18 MIPS at 50 MHz	1988
Namco System 21 (10-processor)	73.927 MIPS at 25 MHz	1988
Intel i386DX	4.3 MIPS at 33 MHz	1989
Intel i486DX	8.7 MIPS at 25 MHz	1989

Processor / System	MIPS	Year
NEC V80	16.5 MIPS at 33 MHz	1989
Intel i860	25 MIPS at 25 MHz	1989
Atari Hard Drivin' (7-processor)	33.573 MIPS at 50 MHz	1989
NEC SX-3 (4-processor)	680 MIPS at 400 MHz	1989
ARM3	12 MIPS at 25 MHz	1989
Motorola 68040	44 MIPS at 40 MHz	1990
Namco System 21 (Galaxian³) (96-processor)	1,660.386 MIPS at 40 MHz	1990
AMD Am386	9 MIPS at 40 MHz	1991
Intel i486DX	11.1 MIPS at 33 MHz	1991
Intel i860	50 MIPS at 50 MHz	1991
Intel i486DX2	25.6 MIPS at 66 MHz	1992
Alpha 21064	86 MIPS at 150 MHz	1992
Alpha 21064	135 MIPS at 200 MHz	1993
MIPS R4400	85 MIPS at 150 MHz	1993
Gmicro/500	132 MIPS at 66 MHz	1993
IBM-Motorola PowerPC 601	157.7 MIPS at 80 MHz	1993
SGI Onyx RealityEngine2 (36-processor)	2,640 MIPS at 150 MHz	1993
Namco Magic Edge Hornet Simulator (36-processor)	2,880 MIPS at 150 MHz	1993
ARM7	40 MIPS at 45 MHz	1994
Intel DX4	70 MIPS at 100 MHz	1994
Motorola 68060	110 MIPS at 75 MHz	1994
Intel Pentium	188 MIPS at 100 MHz	1994
Microchip PIC16F	5 MIPS at 20 MHz	1995
IBM-Motorola PowerPC 603e	188 MIPS at 133 MHz	1995
ARM 7500FE	35.9 MIPS at 40 MHz	1996
IBM-Motorola PowerPC 603ev	423 MIPS at 300 MHz	1996
Intel Pentium Pro	541 MIPS at 200 MHz	1996
Hitachi SH-4	360 MIPS at 200 MHz	1997
IBM-Motorola PowerPC 750	525 MIPS at 233 MHz	1997
Zilog eZ80	80 MIPS at 50 MHz	1999

Multithreading e CPUs Multicores

Processor / System	MIPS	Year
Intel Pentium III	2,054 MIPS at 600 MHz	1999
Sega Naomi Multiboard (32-processor)	6,400 MIPS at 200 MHz	1999
Freescale MPC8272	760 MIPS at 400 MHz	2000
AMD Athlon	3,561 MIPS at 1.2 GHz	2000
Silicon Recognition ZISC 78	8,600 MIPS at 33 MHz	2000
ARM11	515 MIPS at 412 MHz	2002
AMD Athlon XP 2500+	7,527 MIPS at 1.83 GHz	2003
Pentium 4 Extreme Edition	9,726 MIPS at 3.2 GHz	2003
Microchip PIC10F	1 MIPS at 4 MHz	2004
ARM Cortex-M3	125 MIPS at 100 MHz	2004
Nios II	190 MIPS at 165 MHz	2004
MIPS32 4KEc	356 MIPS at 233 MHz	2004
VIA C7	1,799 MIPS at 1.3 GHz	2005
ARM Cortex-A8	2,000 MIPS at 1.0 GHz	2005
AMD Athlon FX-57	12,000 MIPS at 2.8 GHz	2005
AMD Athlon 64 3800+ X2 (2-core)	14,564 MIPS at 2.0 GHz	2005
PowerPC G4 MPC7448	3,910 MIPS at 1.7 GHz	2005
ARM Cortex-R4	450 MIPS at 270 MHz	2006
MIPS32 24K	604 MIPS at 400 MHz	2006
PS3 Cell BE (PPE only)	10,240 MIPS at 3.2 GHz	2006
IBM Xenon CPU (3-core)	19,200 MIPS at 3.2 GHz	2005
AMD Athlon FX-60 (2-core)	18,938 MIPS at 2.6 GHz	2006
Intel Core 2 Extreme X6800 (2-core)	27,079 MIPS at 2.93 GHz	2006
Intel Core 2 Extreme QX6700 (4-core)	49,161 MIPS at 2.66 GHz	2006
MIPS64 20Kc	1,370 MIPS at 600 MHz	2007
P.A. Semi PA6T-1682M	8,800 MIPS at 1.8 GHz	2007
Qualcomm Scorpion (Cortex A8-like)	2,100 MIPS at 1 GHz	2008
Intel Atom N270	3,846 MIPS at 1.6 GHz	2008
Intel Core 2 Extreme QX9770 (4-core)	59,455 MIPS at 3.2 GHz	2008
Intel Core i7 920 (4-core)	82,300 MIPS at 2.93 GHz	2008

Processor / System	MIPS	Year
ARM Cortex-M0	45 MIPS at 50 MHz	2009
ARM Cortex-A9 (2-core)	7,500 MIPS at 1.5 GHz	2009
AMD Phenom II X4 940 Black Edition	42,820 MIPS at 3.0 GHz	2009
AMD Phenom II X6 1100T	78,440 MIPS at 3.3 GHz	2010
Intel Core i7 Extreme Edition 980X (6-core)	147,600 MIPS at 3.33 GHz	2010
ARM Cortex A5	1,256 MIPS at 800 MHz	2011
ARM Cortex A7	2,850 MIPS at 1.5 GHz	2011
Qualcomm Krait (Cortex A15-like, 2-core)	9,900 MIPS at 1.5 GHz	2011
AMD E-350 (2-core)	10,000 MIPS at 1.6 GHz	2011
Nvidia Tegra 3 (Quad core Cortex-A9)	13,800 MIPS at 1.5 GHz	2011
Samsung Exynos 5250 (Cortex-A15-like 2-core)	14,000 MIPS at 2.0 GHz	2011
Intel Core i5-2500K (4-core)	83,000 MIPS at 3.3 GHz	2011
Intel Core i7 875K	92,100 MIPS at 2.93 GHz	2011
AMD FX-8150 (8-core)	90,749 MIPS at 3.6 GHz	2011
Intel Core i7 2600K	117,160 MIPS at 3.4 GHz	2011
Intel Core i7-3960X	176,170 MIPS at 3.3 GHz	2011
AMD FX-8350	97,125 MIPS at 4.2 GHz	2012
AMD FX-9590	115,625 MIPS at 5.0 GHz	2012
Intel Core i7 3770K	106,924 MIPS at 3.9 GHz	2012
Intel Core i7 4770K	133,740 MIPS at 3.9 GHz	2013
Intel Core i7 5960X	298,190 MIPS at 3.5 GHz	2014
Raspberry Pi 2	4,744 MIPS at 1.0 GHz	2014
Intel Core i7 6950X	320,440 MIPS at 3.5 GHz	2016
ARM Cortex A73 (4-core)	71,120 MIPS at 2.8 GHz	2016
AMD Ryzen 7 1800X	304,510 MIPS at 3.7 GHz	2017
Intel Core i7-8086K	221,720 MIPS at 5.0 GHz	2018
Intel Core i9-9900K	412,090 MIPS at 4.7 GHz	2018
AMD Ryzen 9 3950X	749,070 MIPS at 4.6 GHz	2019
AMD Ryzen Threadripper 3990X	2,356,230 MIPS at 4.35 GHz	2020

Multithreading e CPUs Multicores

Medida do desempenho:

IPS = clock x IPC

IPS = sockets x clock x IPC

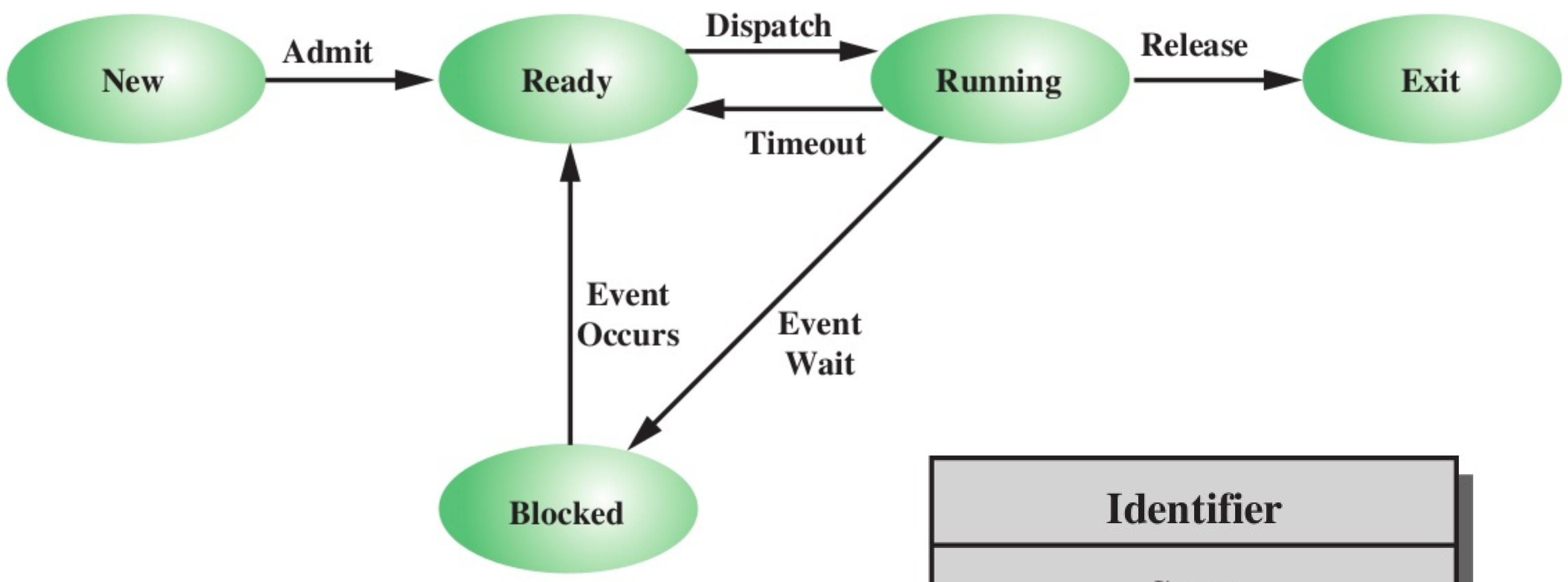
IPS = sockets x cores/socket x clock x IPC

Como melhorar o desempenho?

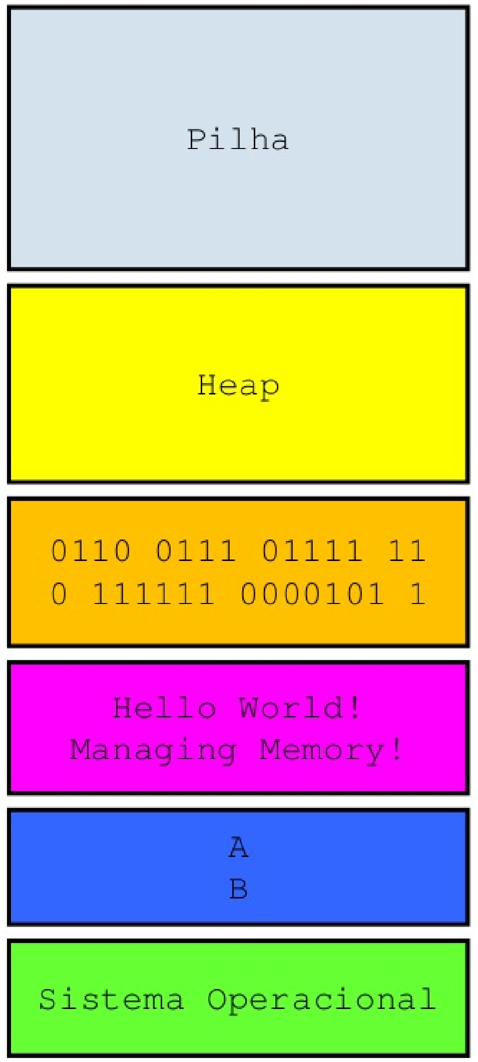
O que são processos?

Processo:

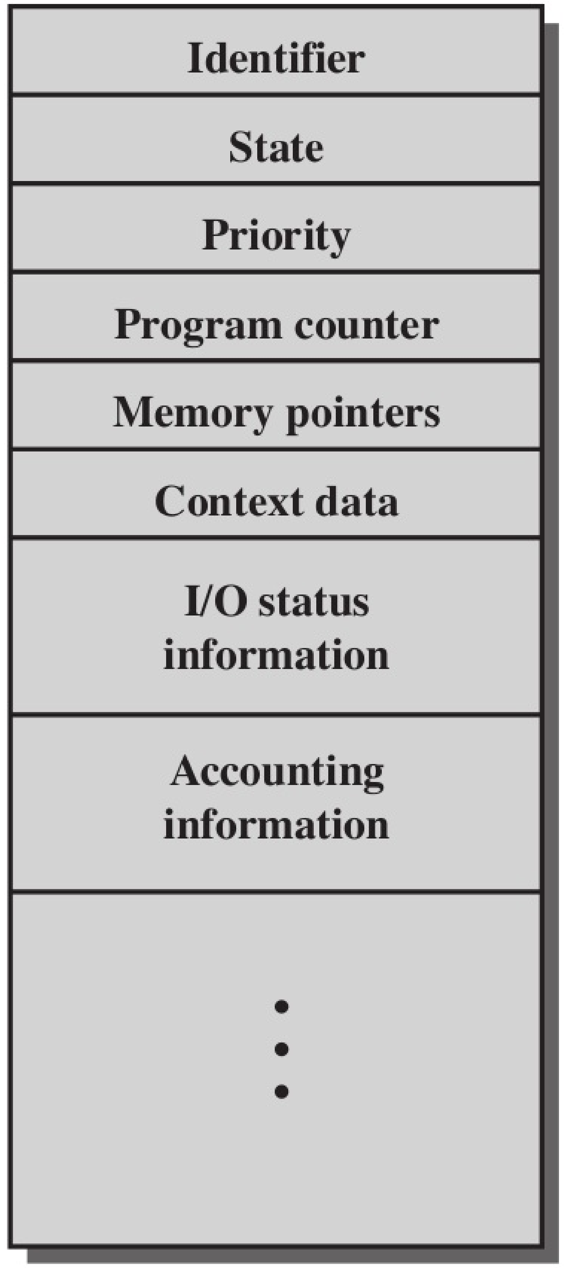
- **é um programa em execução, uma instância de um programa executando no computador**
- **pode ter a posse/controle de recursos do computador (registradores, memória, descritores, I/O, etc...)**
- **pode estar em diversos estados**
- **o SO mantém um "bloco de controle de processo" (PCB) para CADA processo**



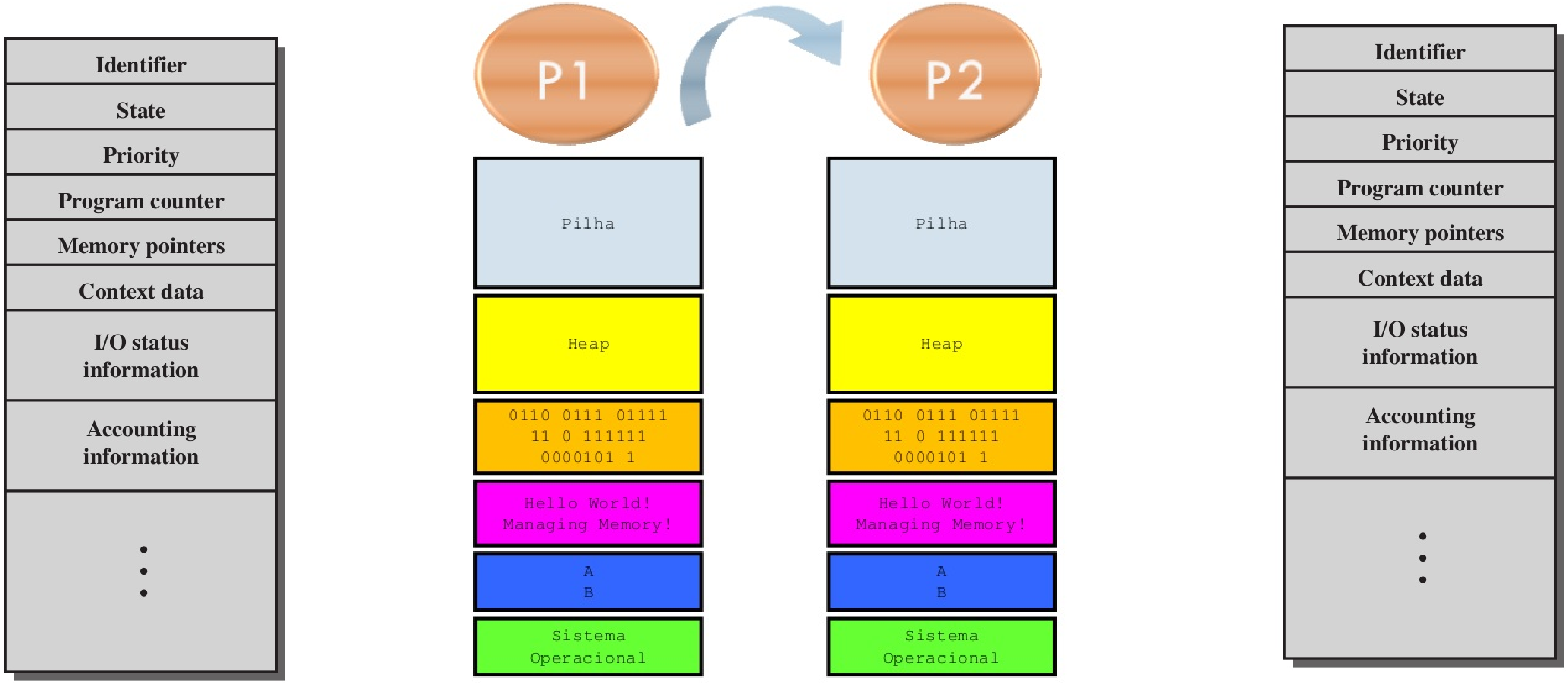
- Processo em memória:
  - **Pilha**:
    - Memória para alocação:
      - Dados de variáveis locais a sub-rotinas
      - Dados do endereço de retorno de uma sub-rotina.
  - **Heap**:
    - Memória para alocação sob-demanda durante a execução
      - Alocação dinâmica
  - **Código Objeto:**
    - Contém as instruções binárias do código executável do processo.
  - **Dados**:
    - Espaço para as variáveis do processo, declaradas como globais no programa.
  - Área do usuário vs. área do sistema.



Espaço de
Endereçamento



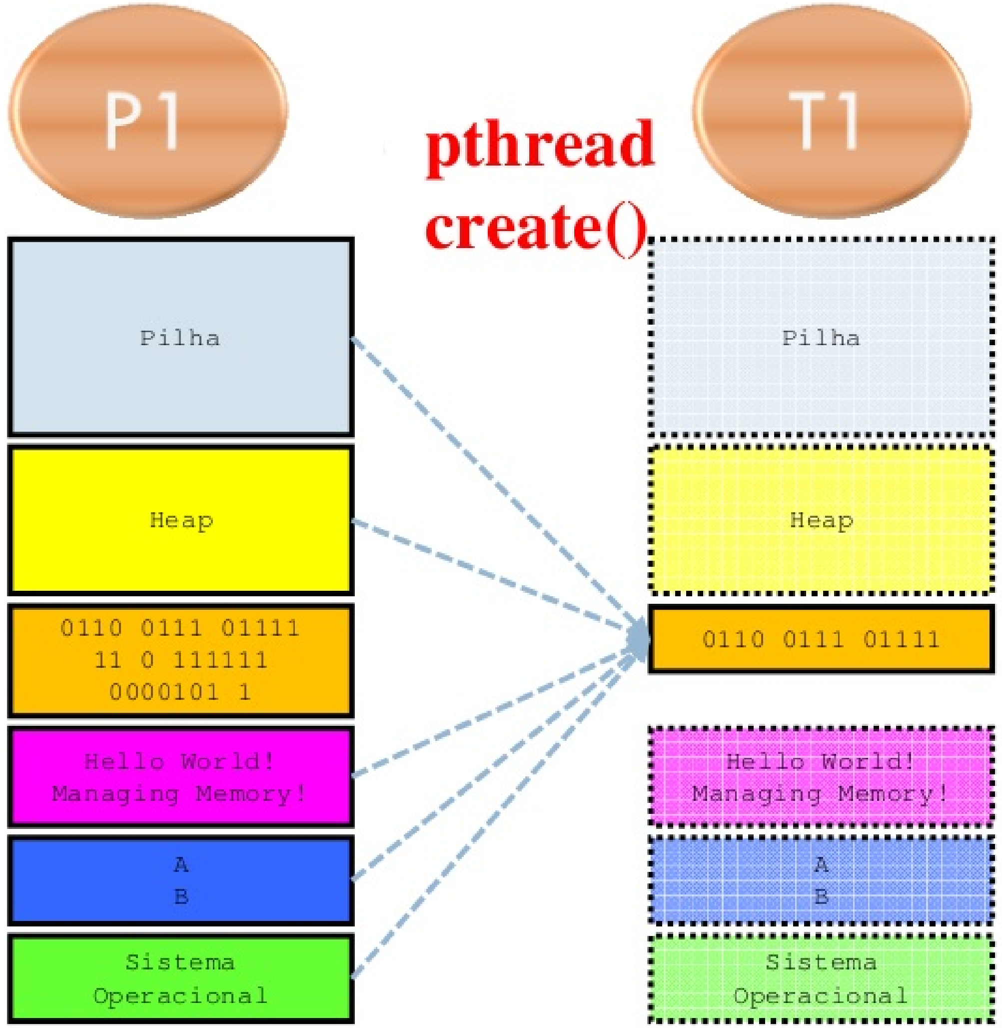
Troca de processos é custosa!



O que são threads?

Thread:

- é uma unidade de trabalho dentro de um processo (seqüência independente de comandos do progama)
- compartilham recursos do processo
- idéia básica: associar mais de um fluxo de execução a um mesmo processo - paralelismo!
- o PCB passa a armazenar a lista de threads do processo

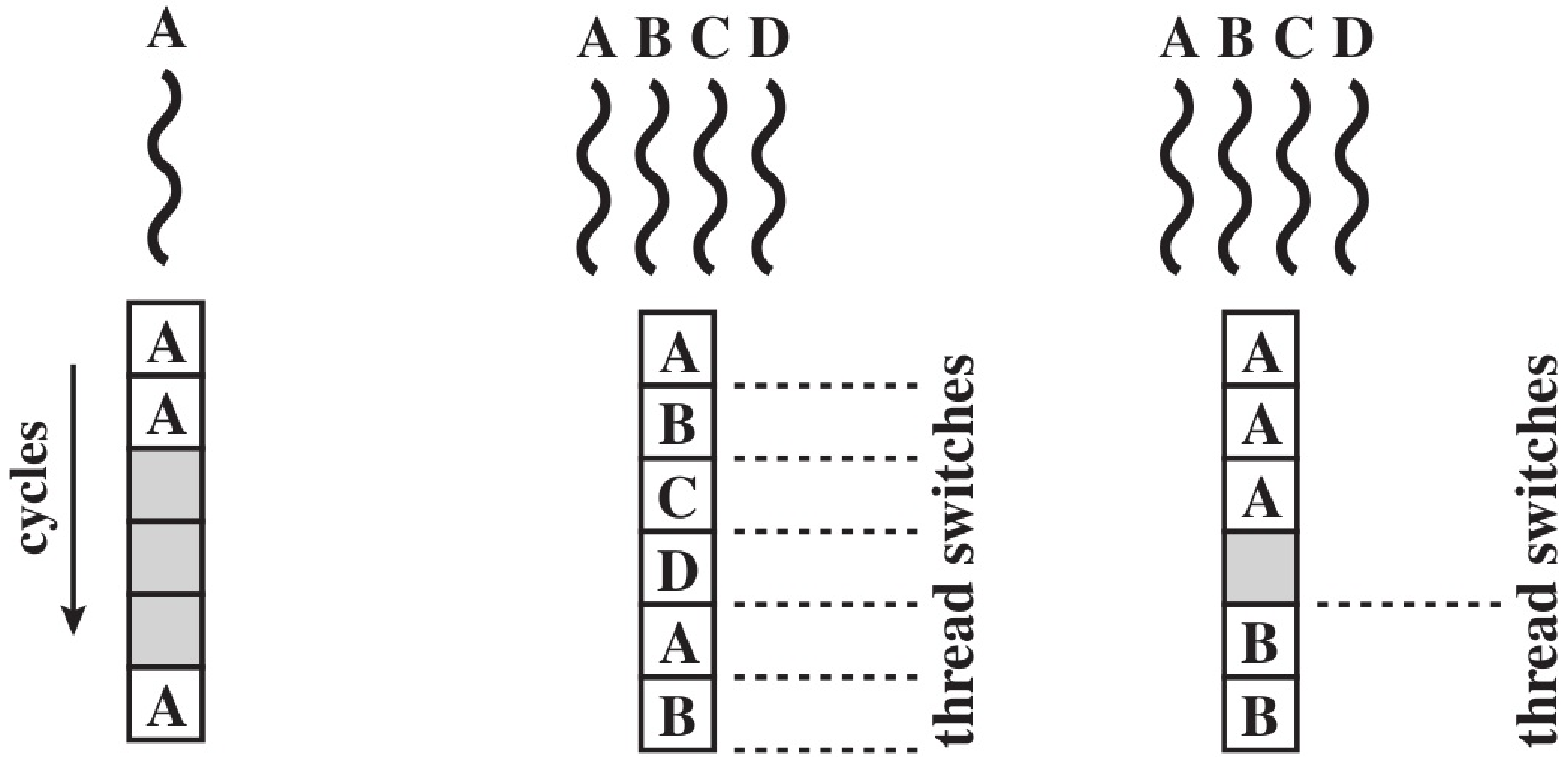


- são "leves"
- troca de threads é rápida
- se a CPU suportar, mais de uma thread pode executar ao mesmo tempo!

Multithreading: abordagens

- Escalar sem multithreading

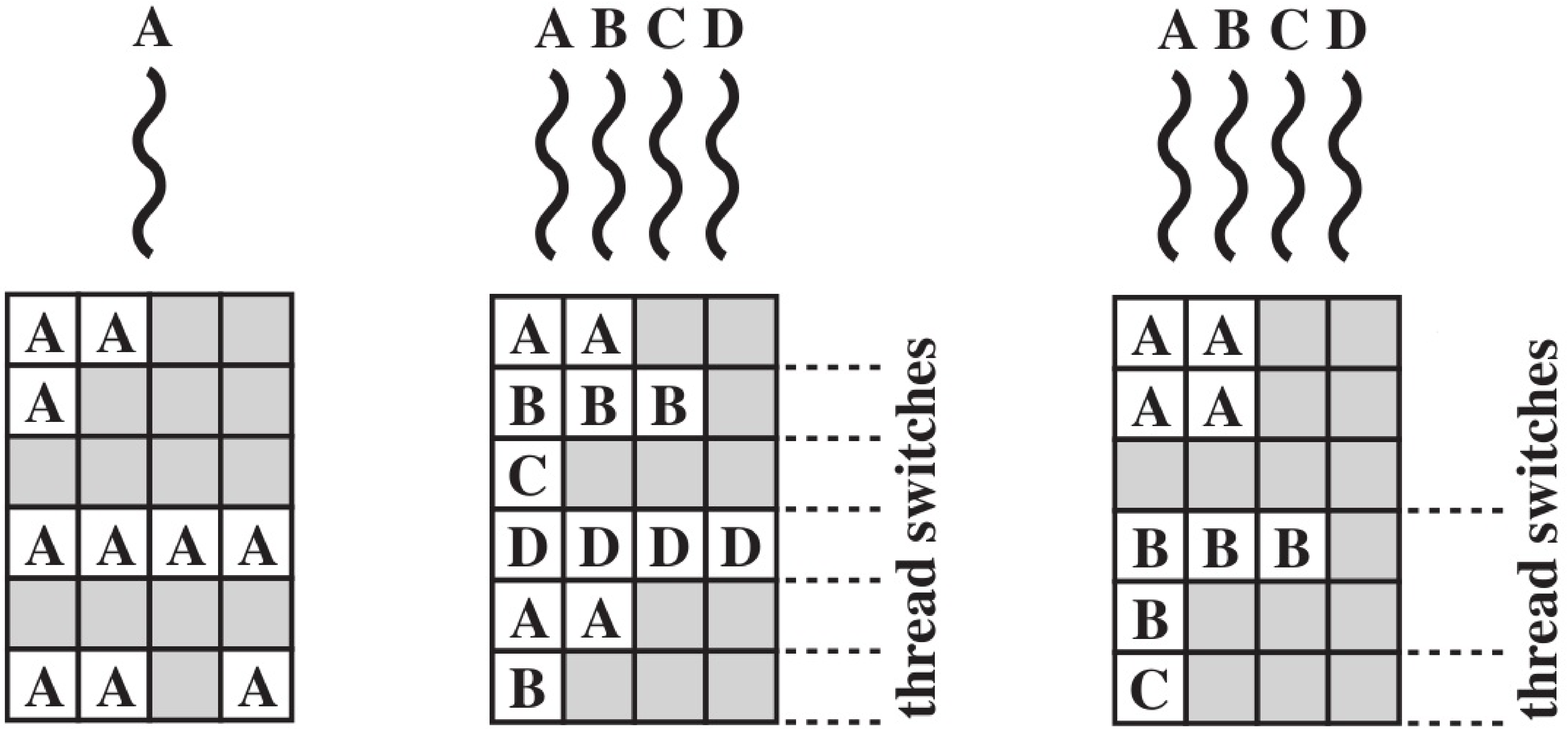
- Multithreading escalar:
    - intercalado
    - bloqueado



Multithreading: abordagens

- **Superescalar sem multithreading:**

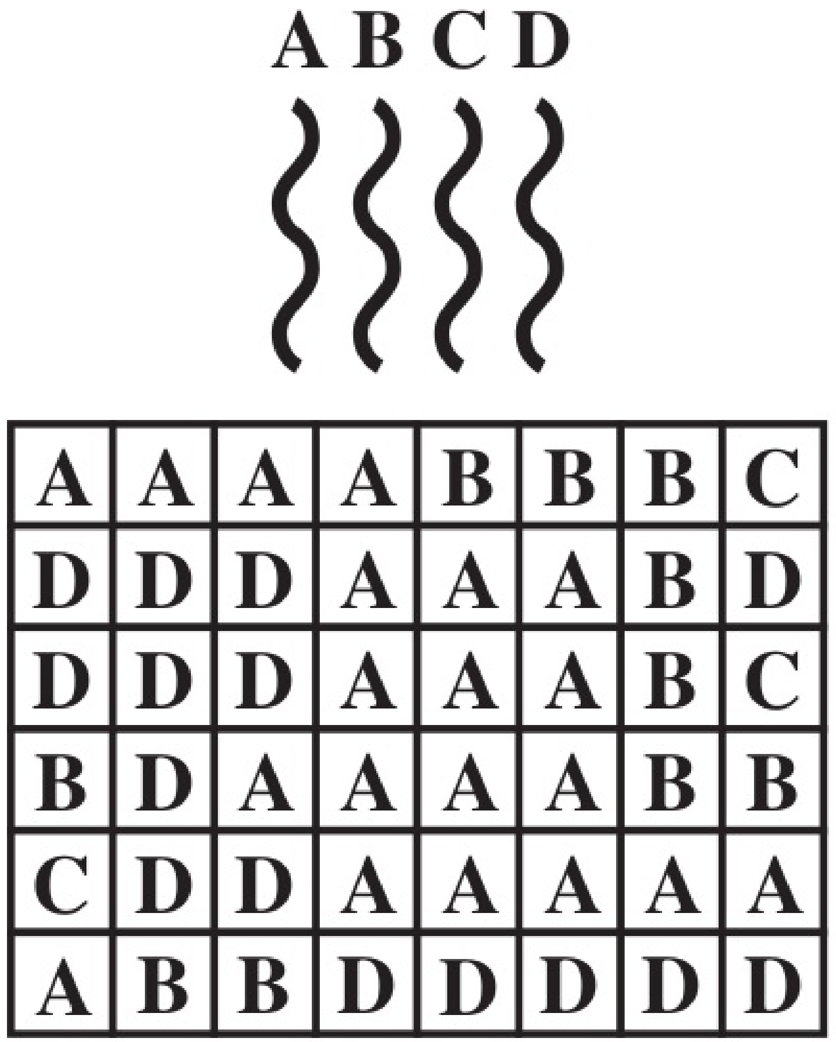
- **Multithreading superescalar:**
    - **intercalado**
    - **bloqueado**

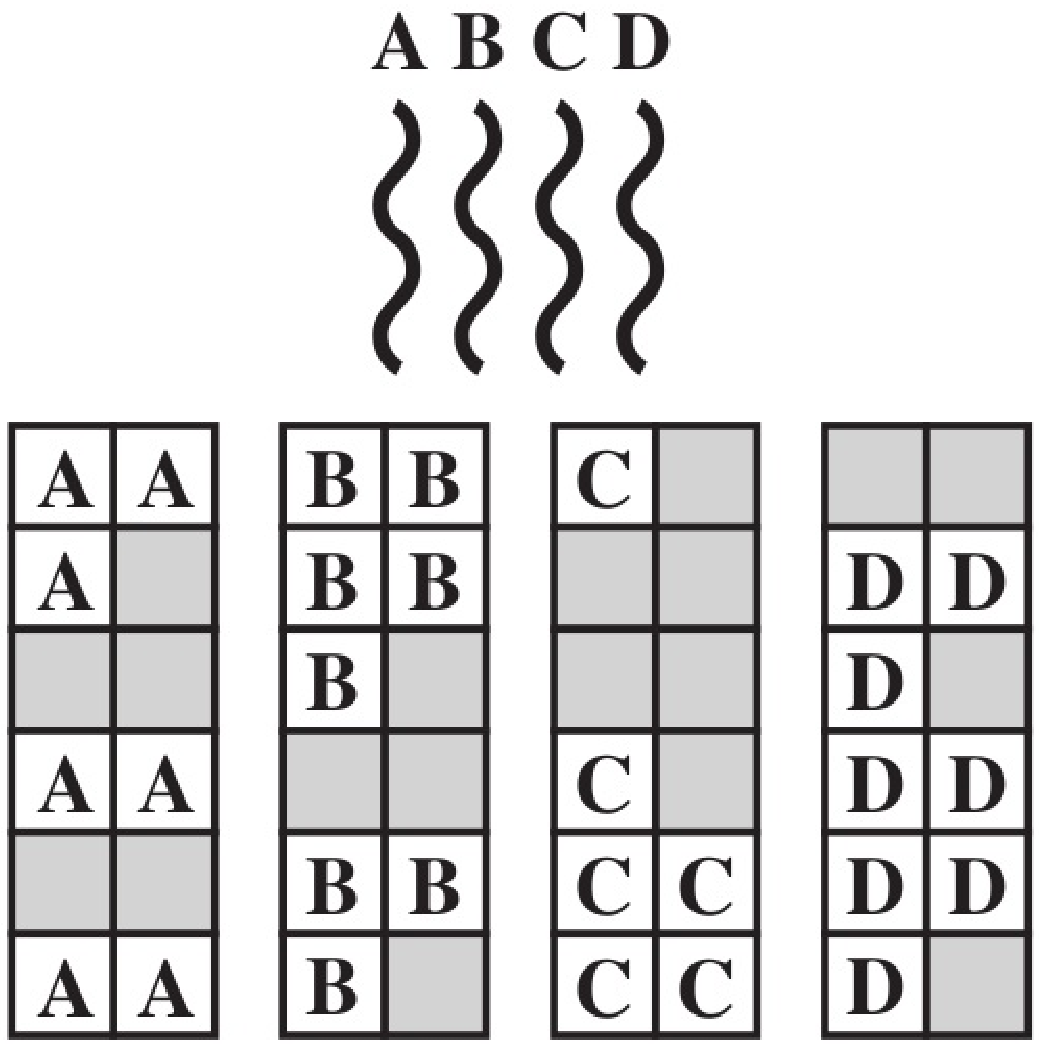


Multithreading: abordagens

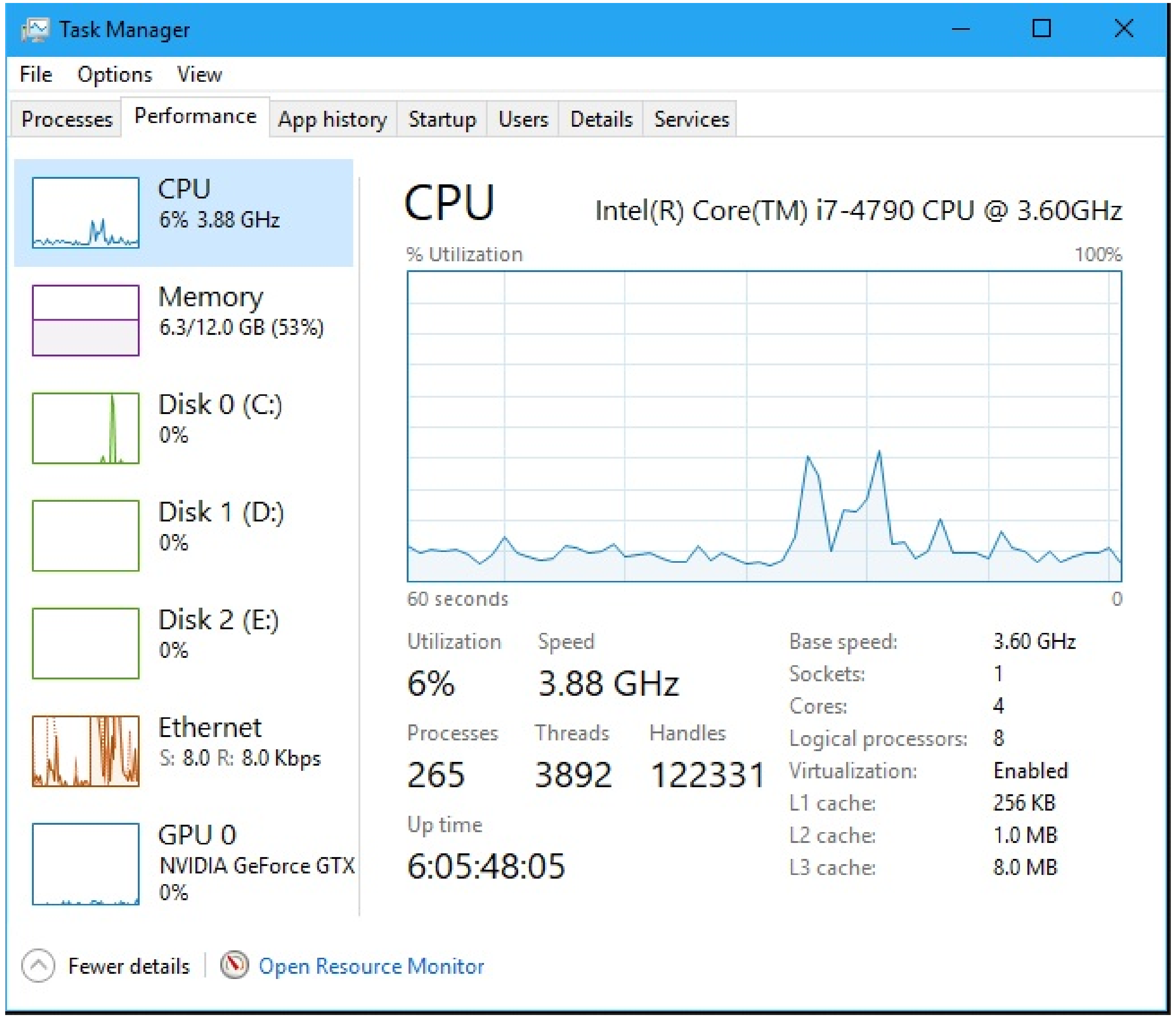
- **Multithreading simultâneo (SMT)**

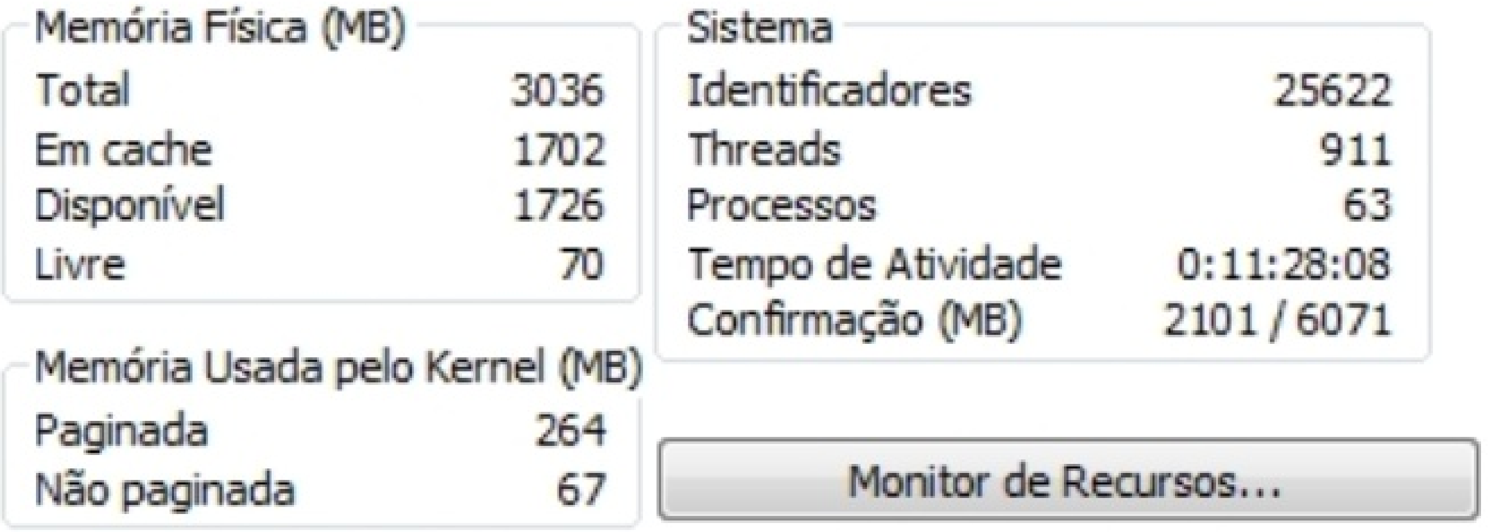
- **Multicores**





"Visualização" de threads





"Visualização" de threads

abrantesasf@abrantes-pc: ~

File Edit View Search Terminal Help

```
top - 16:41:30 up  3:59,  1 user,  load average: 0,50, 0,49, 0,44
Threads: 1801 total,   1 running, 1800 sleeping,   0 stopped,   0 zombie
%Cpu(s):  1,9 us,  1,2 sy,  0,0 ni, 96,9 id,  0,0 wa,  0,0 hi,  0,1 si,  0,0 st
MiB Mem :  64218,5 total,  54329,4 free,   4988,6 used,   4900,5 buff/cache
MiB Swap:   8789,0 total,   8789,0 free,      0,0 used.  58337,8 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
 4331 root      20   0 7054536 220696 181528 S   2,6   0,3   5:25.34 Xorg
 4984 abrante+  -6   0 5678084  23888  18464 S   2,3   0,0   8:08.41 alsa-sink-CA013
26897 abrante+  20   0 2636744 164172 107184 S   2,0   0,2   0:03.38 Isolated Web Co
27150 abrante+  20   0  964952  49944  39392 S   2,0   0,1   0:00.43 mate-terminal
 4704 abrante+  20   0 1618444 186636  97436 S   1,3   0,3   1:42.92 caja
 4711 abrante+  20   0  673644  57816  34248 S   1,3   0,1   1:43.32 wnck-applet
27255 abrante+  20   0   13820   5892   3252 R   1,3   0,0   0:00.47 top
26674 abrante+  20   0 2421604 108816  89760 S   1,0   0,2   0:01.39 Isolated Web Co
 2425 root      20   0   81932   3760   3436 S   0,7   0,0   0:01.00 irqbalance
 4350 root      20   0 7054536 220696 181528 S   0,7   0,3   0:24.36 InputThread
 4332 root     -51   0       0      0      0 S   0,7   0,0   1:12.97 irq/50-nvidia
 4684 abrante+  20   0  693828  40420  28424 S   0,7   0,1   1:31.19 marco
 5148 abrante+  20   0 4959852 800300 297388 S   0,7   1,2   1:10.77 GLXVsyncThread
 5168 abrante+  20   0 4959852 800300 297388 S   0,7   1,2   1:25.29 IPDL Background
26678 abrante+  20   0 2421604 108816  89760 S   0,7   0,2   0:00.62 IPC I/O Child
 4699 abrante+  20   0 1639860  54576  35316 S   0,3   0,1   0:05.18 mate-panel
 5123 abrante+  20   0 4959852 800300 297388 S   0,3   1,2   2:50.72 IPC I/O Parent
 6641 abrante+  20   0 3125580 653724 108720 S   0,3   1,0   8:43.06 Isolated Web Co
 8530 abrante+  20   0 2477040 135448  95140 S   0,3   0,2   0:10.47 Isolated Web Co
 8621 abrante+  20   0 2649284 155820 106972 S   0,3   0,2   1:05.41 Isolated Web Co
26432 abrante+  20   0 2477704 147748  96368 S   0,3   0,2   0:02.17 Isolated Web Co
26999 abrante+  20   0 2410364  94576  80652 S   0,3   0,1   0:00.21 Isolated Web Co
    1 root      20   0  168180  12000   8256 S   0,0   0,0   0:02.44 systemd
    2 root      20   0       0      0      0 S   0,0   0,0   0:00.04 kthreadd
    3 root       0 -20       0      0      0 I   0,0   0,0   0:00.00 rcu_gp
    4 root       0 -20       0      0      0 I   0,0   0,0   0:00.00 rcu_par_gp
    6 root       0 -20       0      0      0 I   0,0   0,0   0:00.00 kworker/0:0H-kblockd
    9 root       0 -20       0      0      0 I   0,0   0,0   0:00.00 mm_percpu_wq
   10 root      20   0       0      0      0 S   0,0   0,0   0:00.34 ksoftirqd/0
   11 root      20   0       0      0      0 I   0,0   0,0   0:09.27 rcu_sched
   12 root      rt   0       0      0      0 S   0,0   0,0   0:00.04 migration/0
   13 root     -51   0       0      0      0 S   0,0   0,0   0:00.00 idle_inject/0
   14 root      20   0       0      0      0 S   0,0   0,0   0:00.00 cpuhp/0
   15 root      20   0       0      0      0 S   0,0   0,0   0:00.00 cpuhp/1
```

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

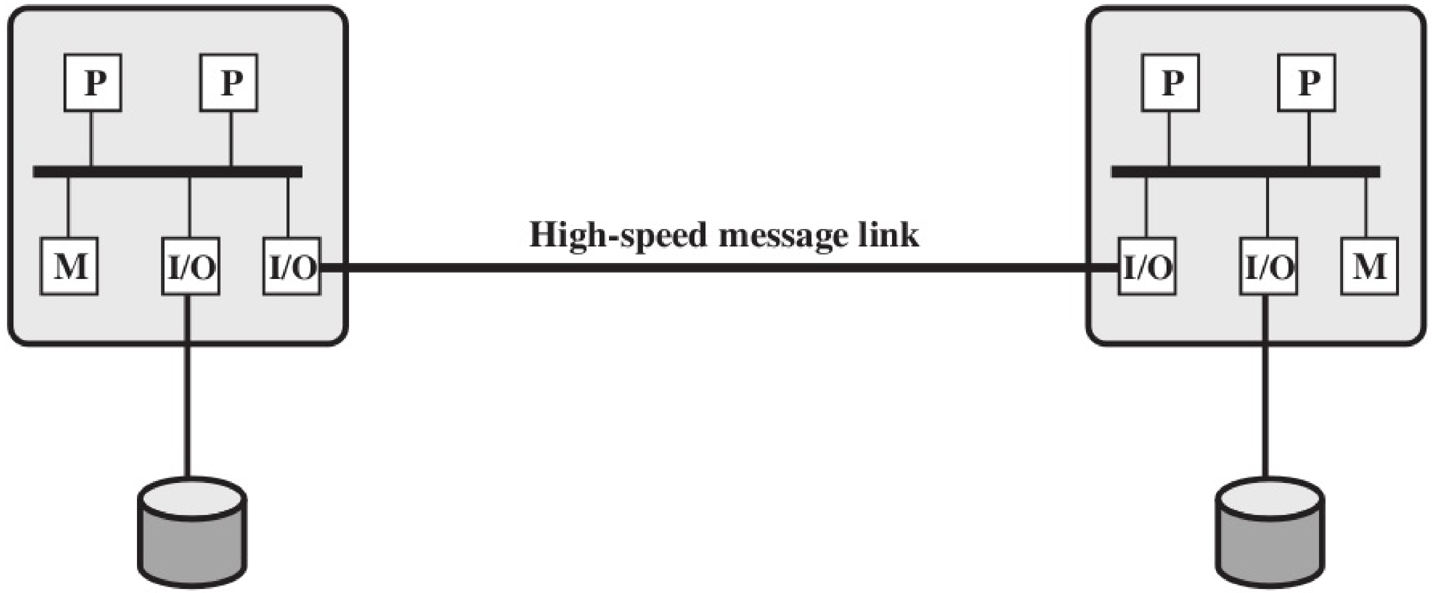
Supercomputadores no Top 500: clusters

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

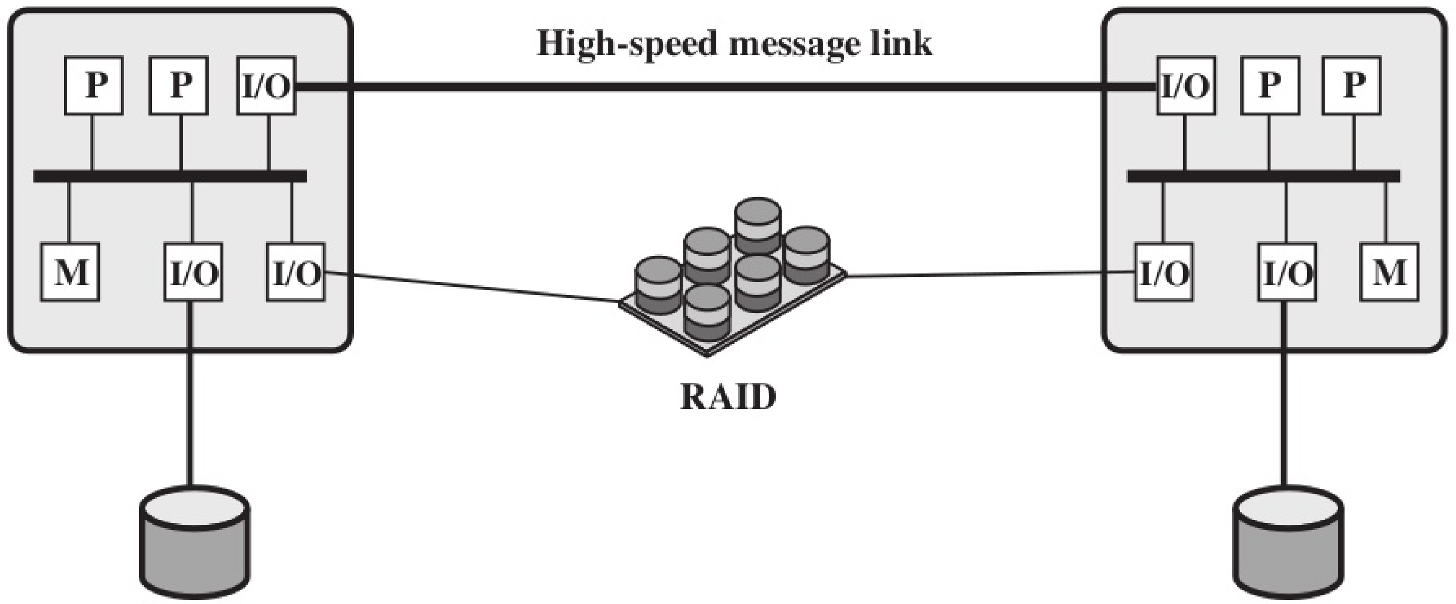
Rank (previous)	Rmax Rpeak (PFLOPS)	Name	Model	CPU cores	Accelerator (e.g. GPU) cores	Interconnect	Manufacturer	Site country	Year	Operating system
1 — (1)	442.010 537.212	Fugaku	Supercomputer Fugaku	158,976 × 48-core Fujitsu A64FX @2.2 GHz	0	Tofu interconnect D	Fujitsu	RIKEN Center for Computational 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



(a) Standby server with no shared disk



(b) Shared disk

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: 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

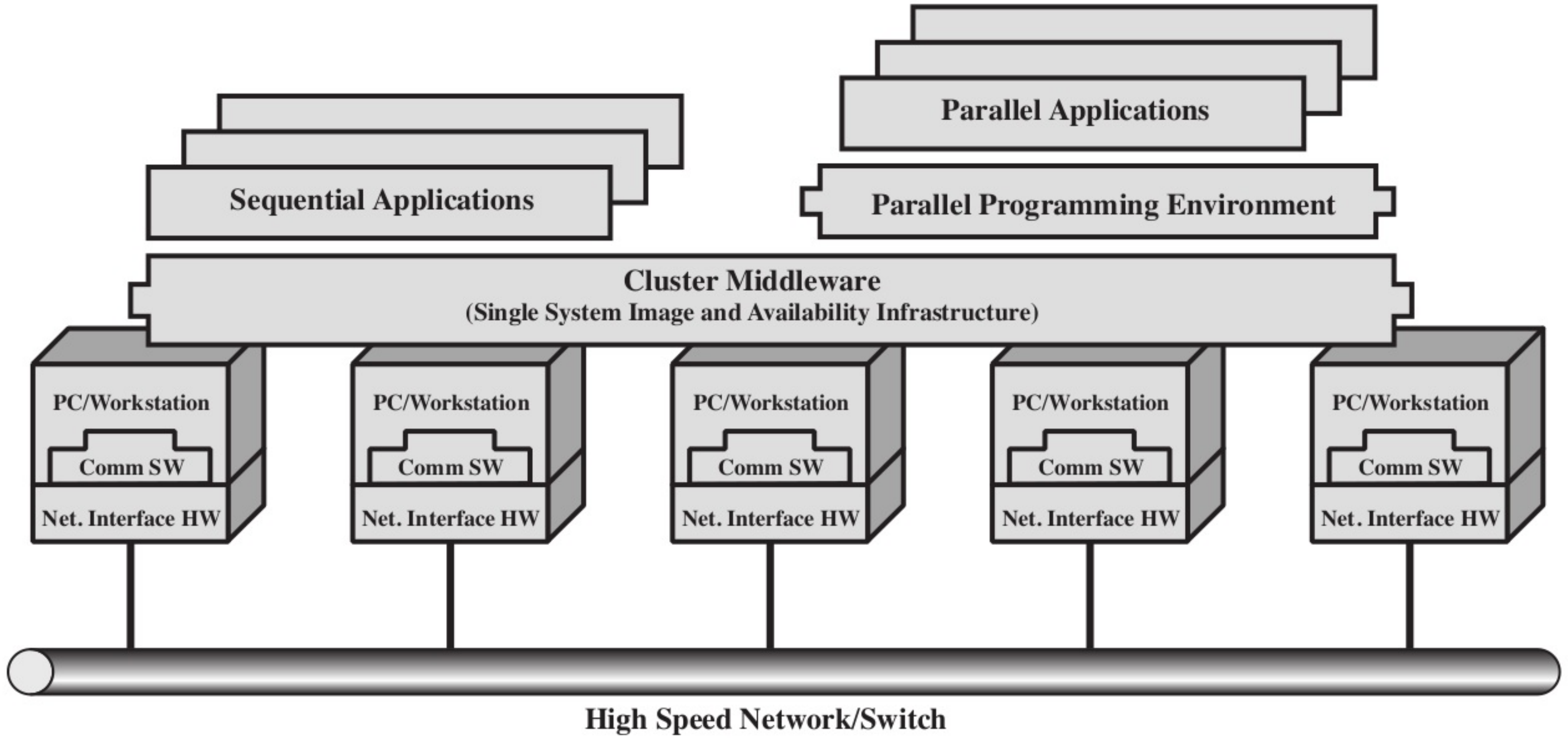
```c
#include <stdio.h>
int main(void) {
        printf("hello, world\n");
        return 0;
}
```

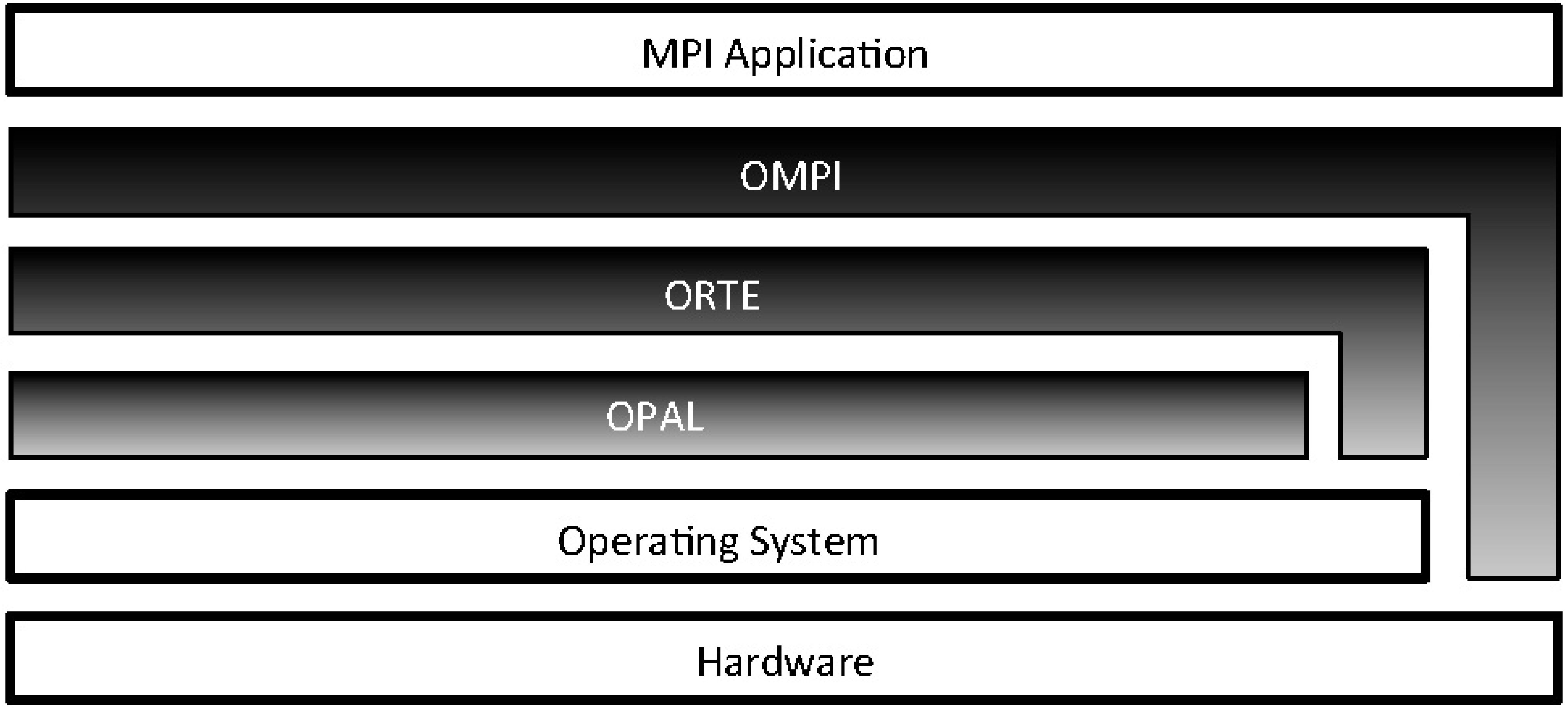
```c
#include <mpi.h>
#include <stdio.h>

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

NSF | TACC | TEXAS

FRONTERA Training User Guide Allocations Fellowships News About Help Search Log in

TACC Frontera User Guide

Search docs

⊟ Notices

- Introduction
- Quickstart
- Account Administration
- Frontera User Portal
- Citizenship on Frontera
- Managing Files
- Launching Applications
- Running Jobs
- Sample Job Scripts
- Job Management
- Building Software
- Programming and Performance
- Visualization on Frontera
- System Architecture
- Cloud Services
- Containers
- Help Desk
- References

Next »

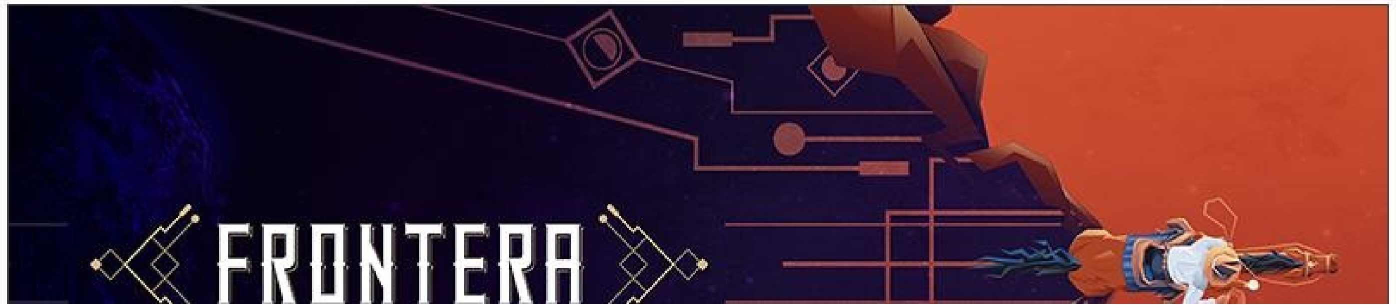
Docs » Notices

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

- 17.4 Multithreading e chips multiprocessadores

- 17.5 Clusters