

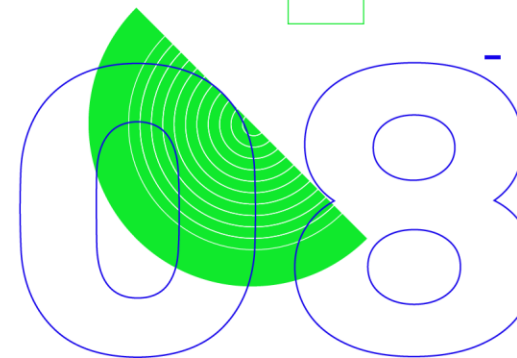
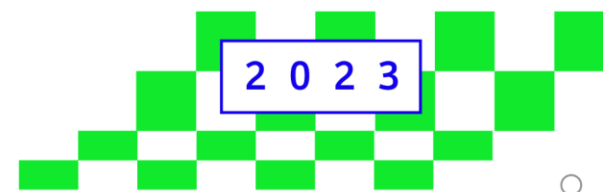
AULA 16

Simulando em *clusters*

Iniciaremos em breve

Código Monte Carlo de interação e transporte de partículas

01
-
02
03
04
-
05
06



01
-
02
03
04
-
05
06

Era uma vez um estudante...

... que resolveu se aventurar no mundo das simulações.



Ele pegou seu notebook e começou a realizar simulações ...



... mas ele só conseguiu fritar um ovo no seu notebook.



Era uma vez um estudante...

... que resolveu se aventurar no mundo das simulações.

Então seu orientador deu a ele um desktop!



... mas ele só conseguiria resultados confiáveis em 30 anos.

Era uma vez um estudante...

... que resolveu se aventurar no mundo das simulações.

Então ele resolveu utilizar os computadores dos colegas de sala...



... agora ele conseguiria resultados confiáveis em 10 anos.

Cluster

Um **cluster** é composto de uma série de computadores conectados por uma rede de modo a realizarem tarefas de processamento em conjunto.



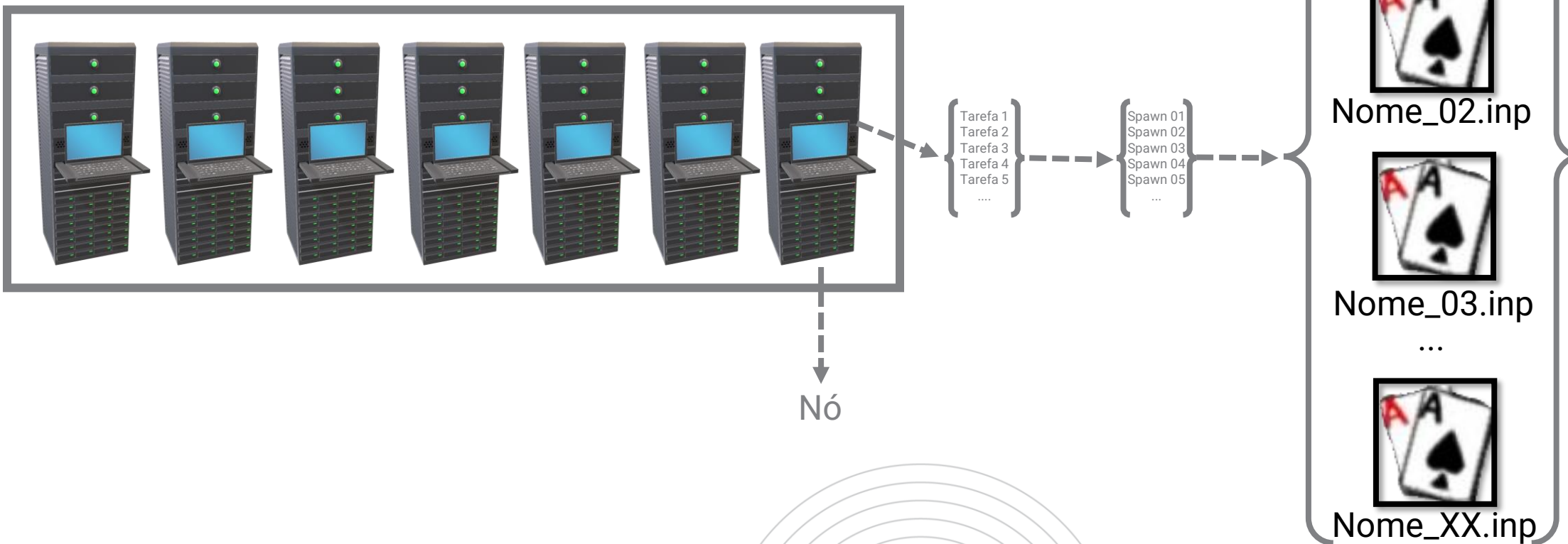
Cluster

Exemplos de instalações com supercomputadores: Santos Dumont (SDumont) em Petrópolis-RJ e CENAPAD-SP/Unicamp em Campinas – SP.



- FLUKA não é um código paralelizável;
- Alternativa: gerar simulações com diferentes sementes e unir os resultados obtidos.

Cluster



Arquivos para submissão em cluster

- Supercomputadores possuem gerenciadores de recursos para realização dos processos (jobs) com linguagem própria;
- Exemplos SLURM (Simple Linux Utility for Resource Management), PBS (Portable Batch System), Moab HPC Suite, Maui Cluster Scheduler e TORQUE (Tera-scale Open-source Resource and QUEue manager).

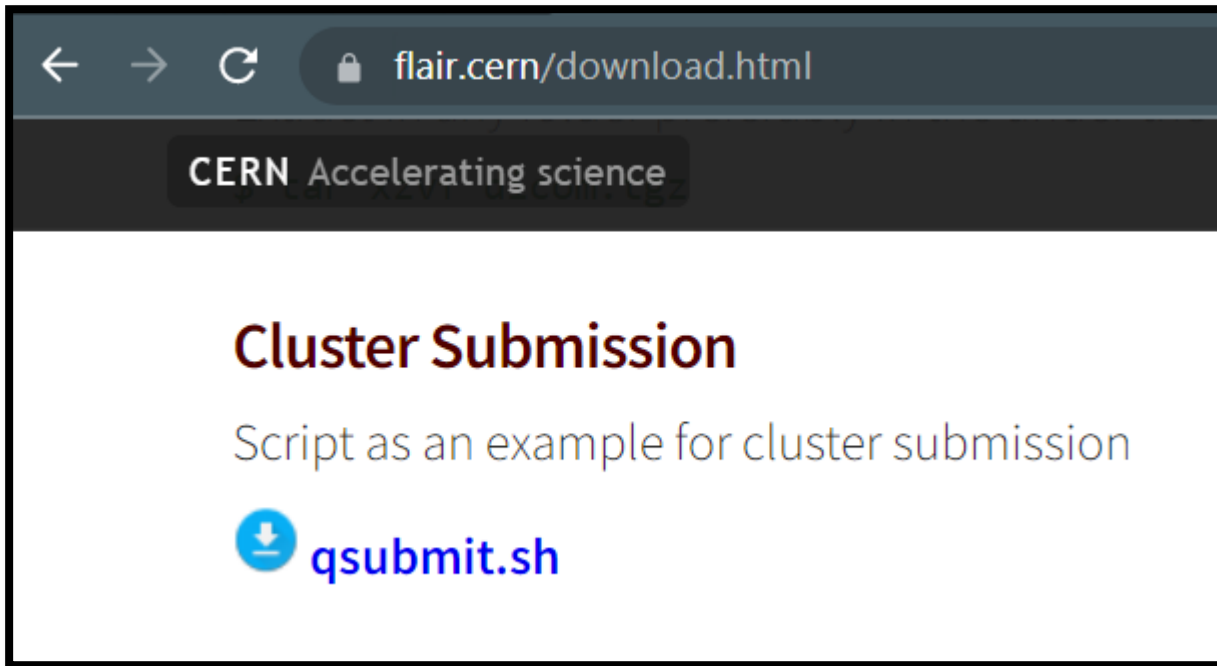


PBS Works™



Adaptive
COMPUTING

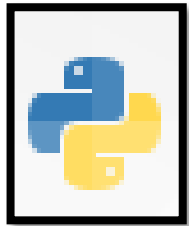
Arquivos para submissão em cluster



A screenshot of a web browser window. The address bar shows "flair.cern/download.html". The page header says "CERN Accelerating science". The main content area has the heading "Cluster Submission" and the subtitle "Script as an example for cluster submission". Below this, there is a blue download icon followed by the text "qsubmit.sh".

```
$ qsubmit.sh X
$ qsubmit.sh
1  #!/bin/bash
2  # Script to launch flair-FLUKA jobs on a cluster using OpenPBS
3  # Author: Vasilis.Vlachoudis@cern.ch
4  # Date: Nov-2006
5
6  NAME="flair${!#}"
7  JOBFILe=${NAME}.job
8  NAME=`echo ${NAME} | cut -c 1-15`
9  if [ -z "${FLUKAPATH}" ]; then
10 |     FLUKAPATH=$PWD
11  fi
12
13  cat > ${JOBFILE} << EOF
14  #!/bin/bash
15  #PBS -q long
16  #PBS -N ${!#}
17  export FLUKAPATH=$FLUKAPATH
18  cd $PWD
19  $*
20  EOF
21
22  echo "Submitting job ${JOBFILE}"
23  chmod +x ${JOBFILE}
24  qsub ${JOBFILE}
25
```

Arquivos para submissão em cluster

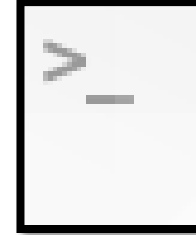


Rotina em python



Arquivos .inp

+






Arquivo .sh

+





Arquivos auxiliares exec.sh

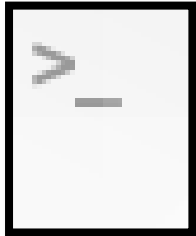
 parfluka_teste_LOVELACE.py	2,4 kB
 teste.flair	34,5 kB
 teste.inp	34,2 kB

Rotina em python



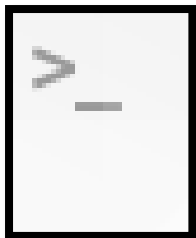
 exec_fluka_N01.sh	110 bytes
 parfluka_teste_LOVELACE.py	2,4 kB
 teste.flair	34,5 kB
 teste.inp	34,2 kB
 teste.sh	755 bytes
 teste_01.inp	34,2 kB
 teste_02.inp	34,2 kB

Arquivos para submissão em cluster



Arquivo
.sh

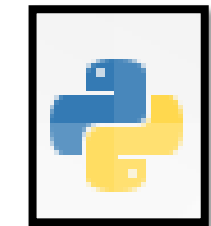
```
teste.sh x
1 #!/bin/bash
2 #SBATCH --nodes=1           #Numero de Nos
3 #SBATCH --ntasks-per-node=2 #Numero de tarefas por No
4 #SBATCH --ntasks=2         #Numero total de tarefas
5 #SBATCH -p sequana_cpu_dev #Fila (partition) a ser utilizada
6 #SBATCH -J teste           #Nome job
7 #SBATCH --exclusive        #Utilização exclusiva dos nos durante a execucao do job
8 #SBATCH --mail-type=ALL
9 #SBATCH --mail-user=isabela.moraes@cnpem.br
10
11 #Exibe os nos alocados para o Job
12 echo $SLURM_JOB_NODELIST
13 nodeset -e $SLURM_JOB_NODELIST
14
15 cd $SLURM_SUBMIT_DIR
16
17 export FLUPRO=/scratch/app/fluka/4-2.1-gfor7
18 module load fluka/4-2.1-gfor7
19 srun --resv-ports --nodes 1 --ntasks=1 -c 24 bash ${PWD}/exec_fluka_N01.sh &
20 wait
21
```



Arquivos auxiliares
exec.sh

```
teste.sh x exec_fluka_N01.sh x
1 #!/bin/bash
2
3 $FLUPRO/bin/rfluka -M 5 ${PWD}/teste_01.inp &
4 $FLUPRO/bin/rfluka -M 5 ${PWD}/teste_02.inp &
5 wait
6
```

Arquivos para submissão em cluster

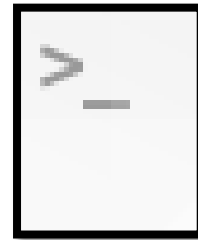


Rotina em python



Arquivos .inp

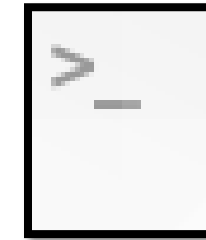
Execução python



× nós

Arquivos auxiliares
exec.sh

+



Arquivo
.sh

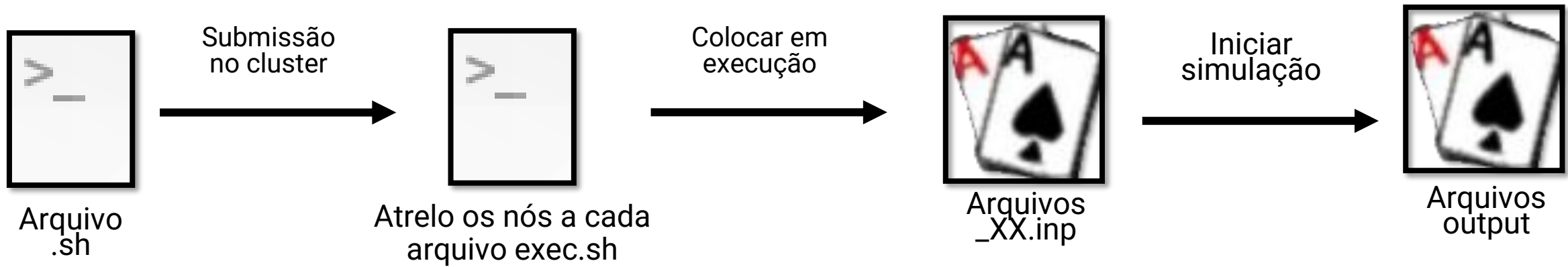
+



× nós × tarefas

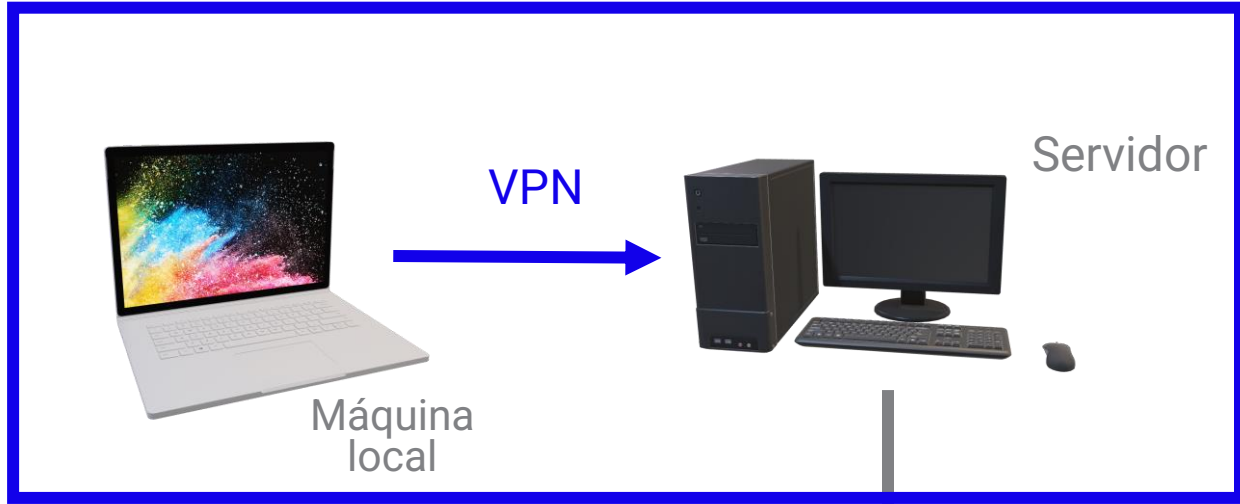
Arquivos
_XX.inp

Arquivos para submissão em cluster



Acesso

Máquina local



Cluster



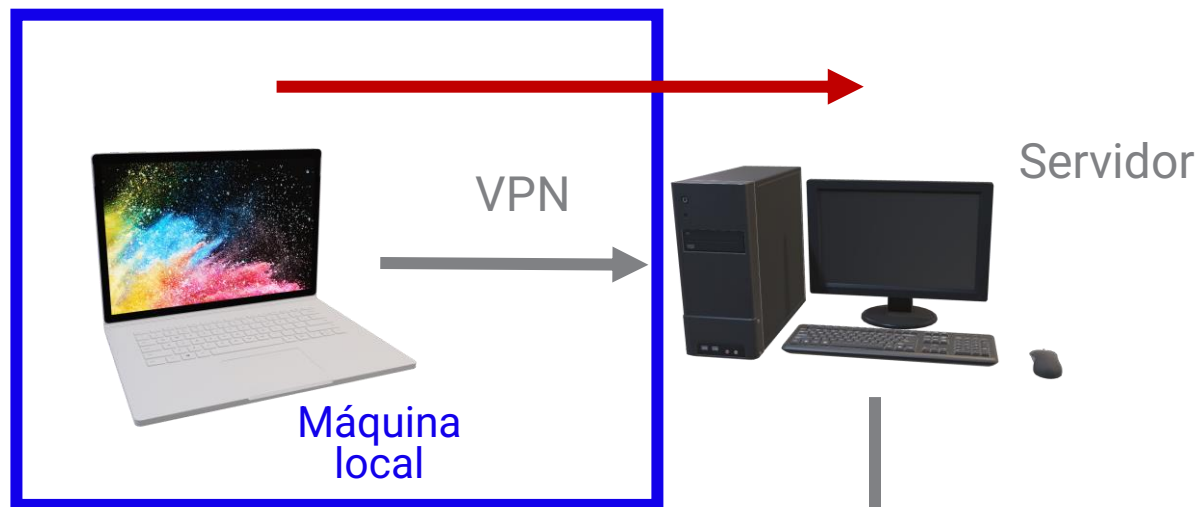
Acesso dado via VPN com:

- Windows com softwares como VPN Client para acesso com interface gráfica.
- Linux com comando ssh no terminal de comando da forma:

```
$ ssh usuário@dominio
```

Envio de arquivos

Máquina local



Cluster



Com o VPN ligado, com:

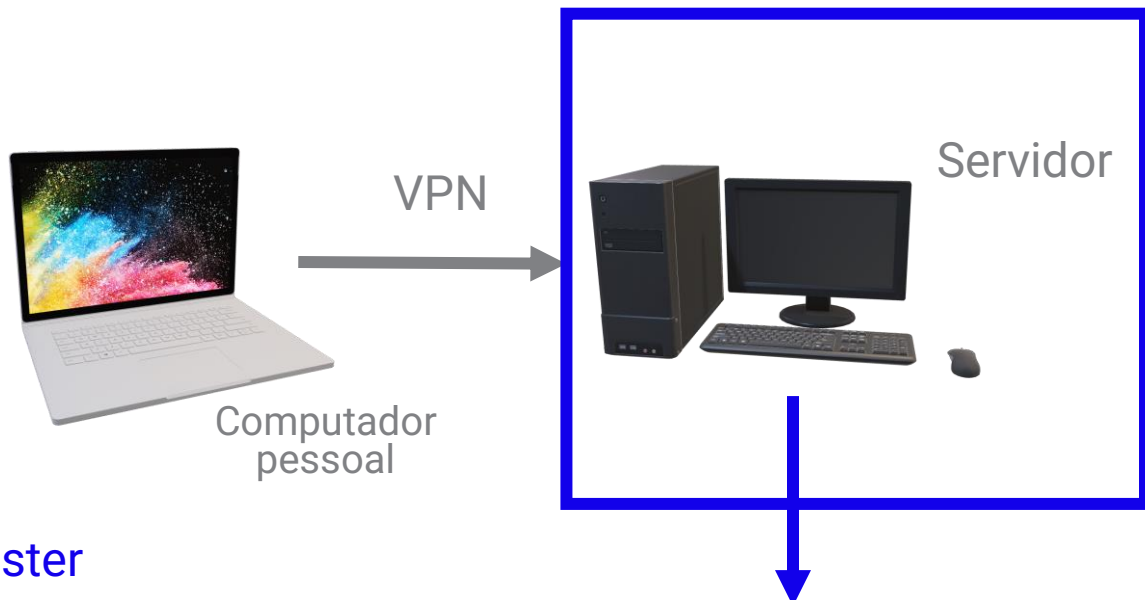
- Linux abrir o terminal de comando da máquina pessoal e usar comandos como:

```
$ scp -r pasta_origem usuário@domínio:/  
pasta_destino
```

```
$ rsync -r pasta_origem usuário@domínio:/  
pasta_destino
```

-r : necessário para envio de pasta. Para enviar um arquivo, basta removê-lo.

Submetendo um job



Cluster



Com VPN e tendo acessado a máquina do supercomputador, você irá utilizar como se fosse uma máquina local.

O ambiente dos dois supercomputadores que utilizamos são máquinas Ubuntu. Portanto, a simulação pode ser submetida via linha de comando.

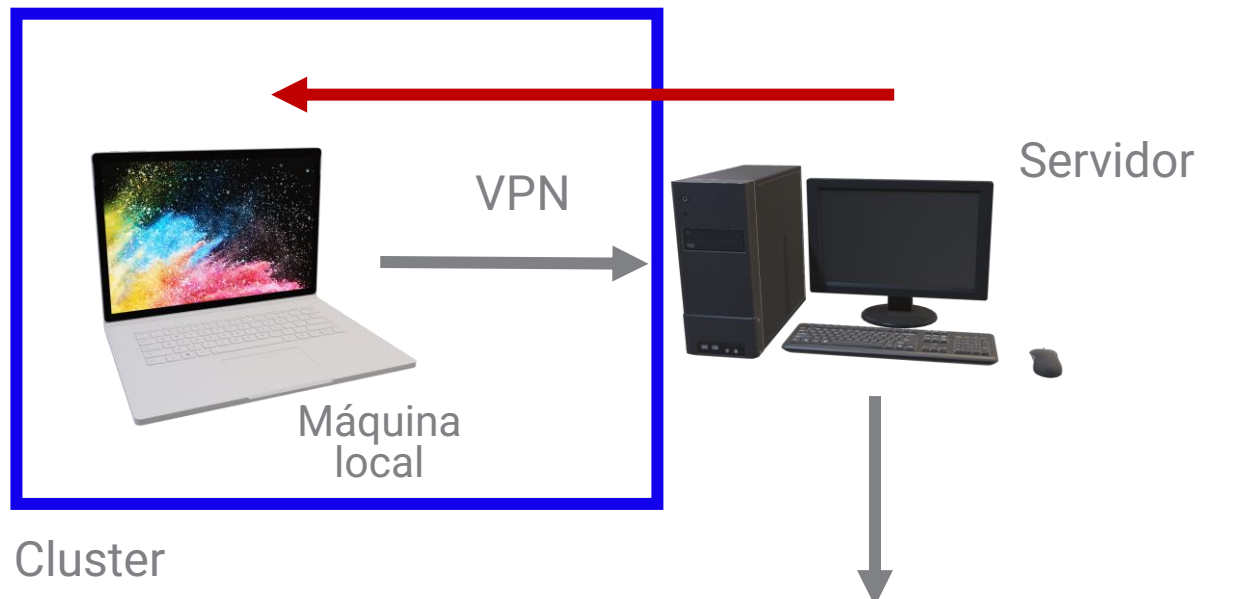
Comandos para submissão de Jobs:

PBS: `$ qsub nome.sh`

SLURM: `$ sbatch nome.sh`

Baixar arquivos

Máquina local



Cluster



Com o VPN ligado, com:

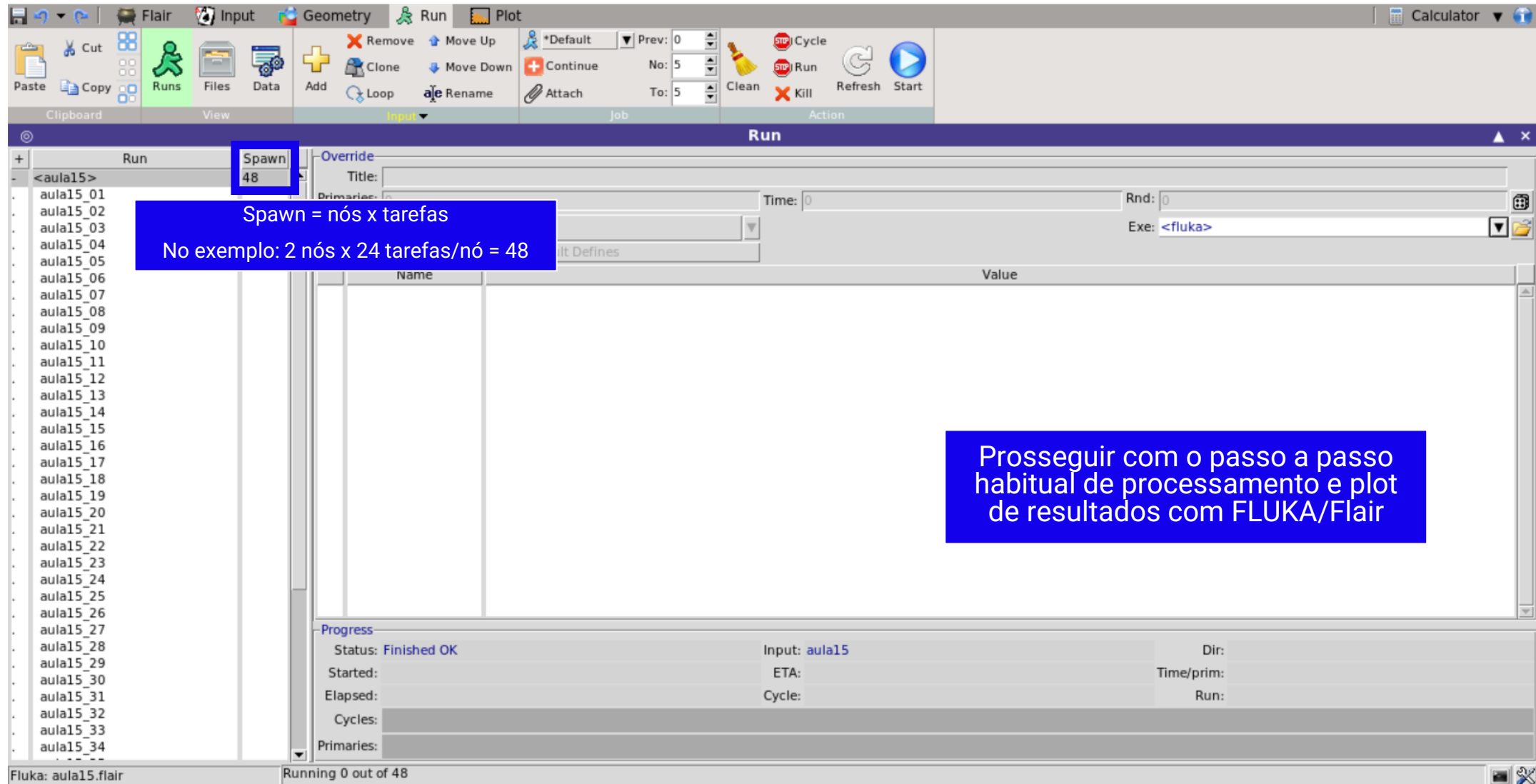
- Linux abrir o terminal de comando da máquina pessoal e usar comandos como:

```
$ scp -r usuário@domínio:/pasta_origem pasta_destino
```

```
$ rsync -r usuário@domínio:/pasta_origem pasta_destino
```

-r : necessário para envio de pasta. Para enviar um arquivo, basta removê-lo.

Unindo resultados



Spawn = nós x tarefas
No exemplo: 2 nós x 24 tarefas/nó = 48

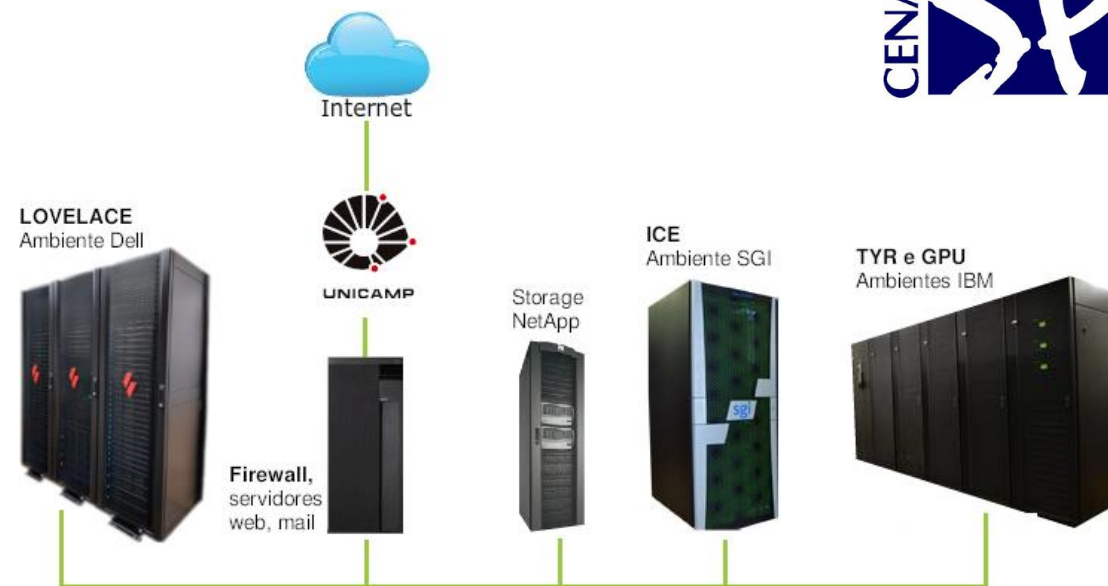
Prosseguir com o passo a passo habitual de processamento e plot de resultados com FLUKA/Flair

Fluka: aula15.flair Running 0 out of 48

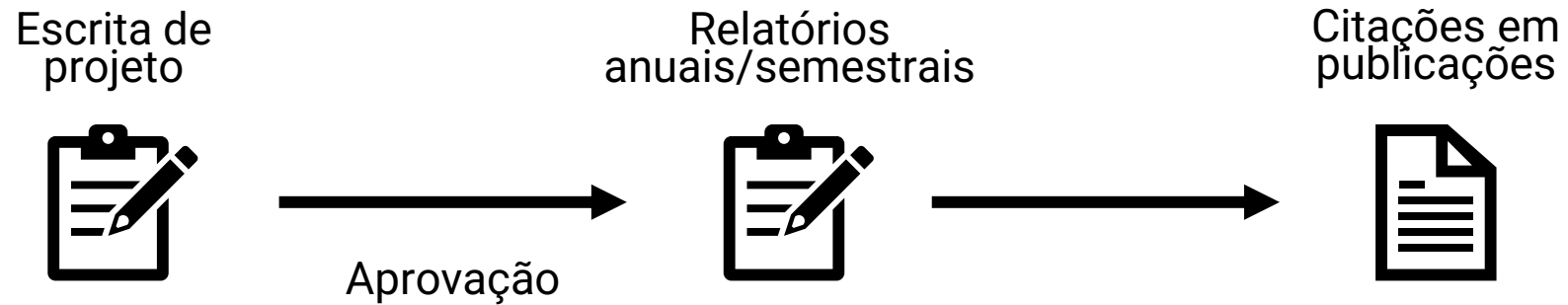
Projeto em supercomputadores



- 504 nós de computação B710 (*thin node*), onde cada nó possui 2 CPU Intel Xeon E5-2695v2 Ivy Bridge (12c @2,4GHz) e 64Gb de memória RAM
- 198 nós de computação B715 (*thin node*) com GPUs K40, onde cada nó possui 2 x CPU Intel Xeon E5-2695v2 Ivy Bridge (12c @2,4GHz) e 64Gb de memória RAM
- 54 nós de computação B715 (*thin node*) com XEON PHI, onde cada nó possui 2 x CPU Intel Xeon E5-2695v2 Ivy Bridge (12c @2,4GHz) e 64Gb de memória RAM
- 1 nó de computação MESCA 2 com memória compartilhada, 16 x CPU Intel Xeon Ivy Bridge (15c @2,4GHz) e 6TB de memória RAM
- 246 nós computacionais (CPU), cada um com 2x Intel Xeon Cascade Lake Gold 6252 e 384Gb de memória RAM
- 36 nós computacionais (CPU), cada um com 2x Intel Xeon Cascade Lake Gold 6252 e 768 Gb de memória RAM
- 94 nós computacionais (GPU), cada um com 2x Intel Xeon Cascade Lake Gold 6252, 4x NVIDIA Volta V100 GPU e 384Gb de memória RAM
- 1 nó para Inteligência Artificial com 2x Intel Xeon Skylake Gold 6148 (20c @2,4GHz), 8x NVIDIA Tesla V100-16GB com NVLink e 384Gb de memória RAM



Projeto em supercomputadores



CURSO INTRODUTÓRIO



23 DE JANEIRO
A 8 DE MARÇO
DE 2023

AULA 16

Simulando em *clusters*

Obrigada pela participação

Código Monte Carlo de interação e transporte de partículas

