

## Big Data – TP1 Part 0

# Connection to the DCE clusters of CentraleSupélec using *dcejs* or *ssh* or *vscode* (Data Center for Education)

Stéphane Vialle

&

Gianluca Quercini



ÉCOLE DOCTORALE  
Sciences et technologies  
de l'information  
et de la communication (STIC)

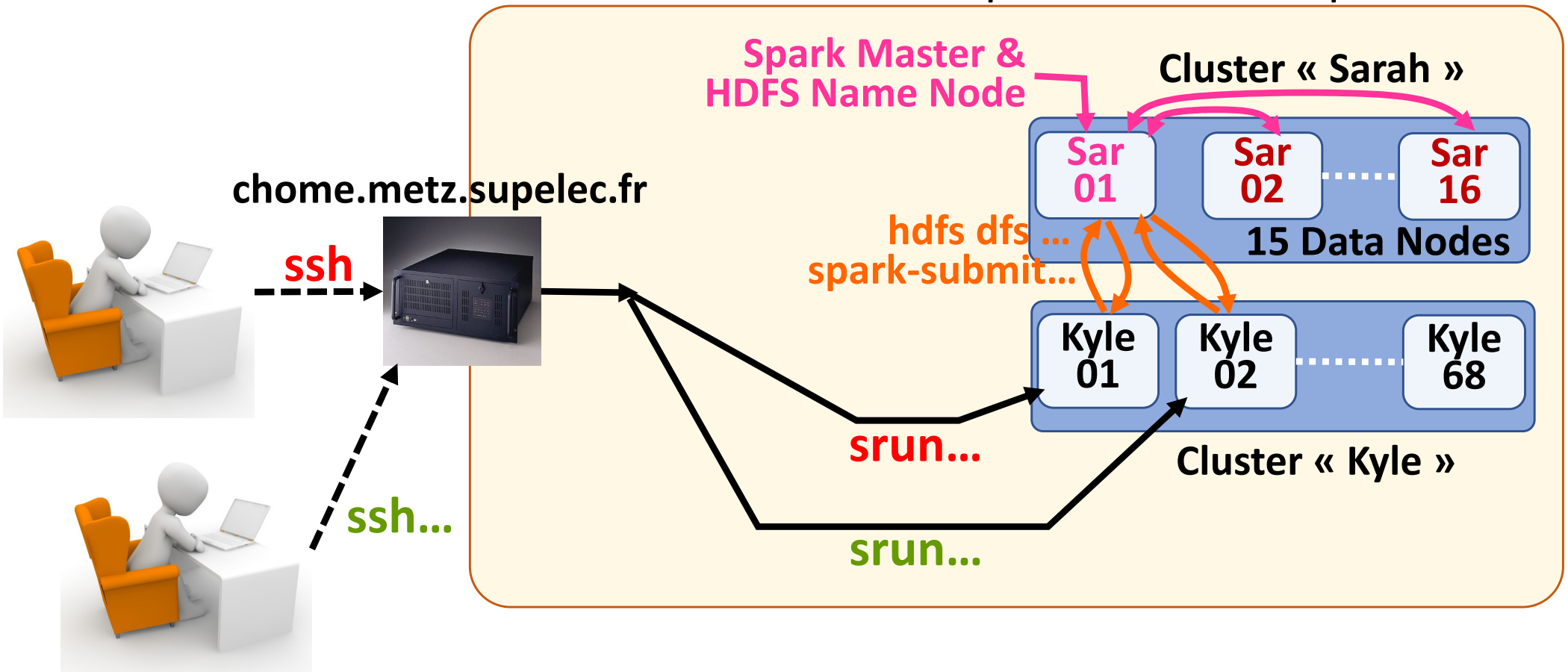


# Using Spark cluster of CentraleSupelec DCE

- **DCE architecture and access with dcejs**
- DCE architecture and access with ssh
- DCE access with vscode
- How to kill a zombie session

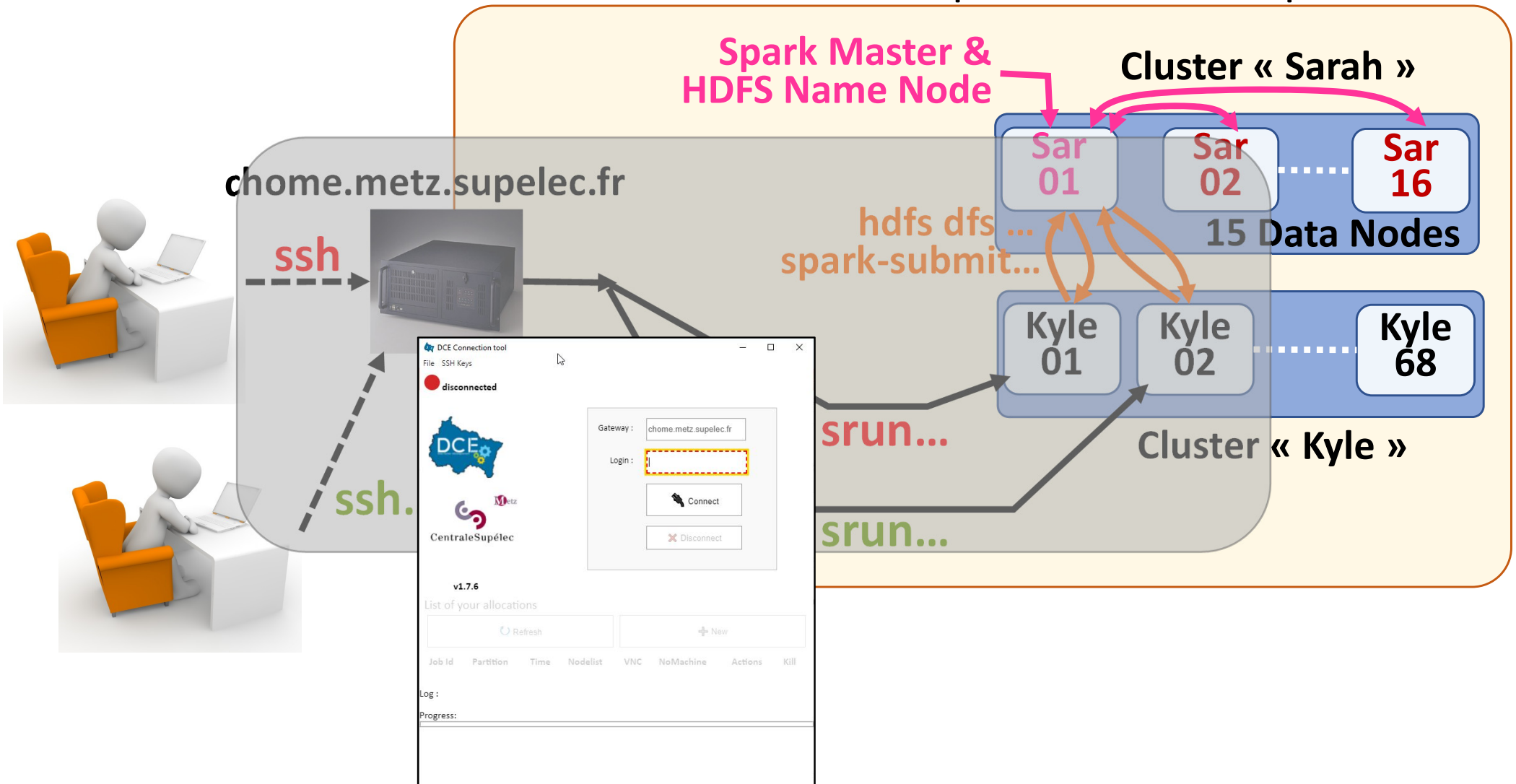
# CPU clusters

## CentraleSupelec Metz Campus



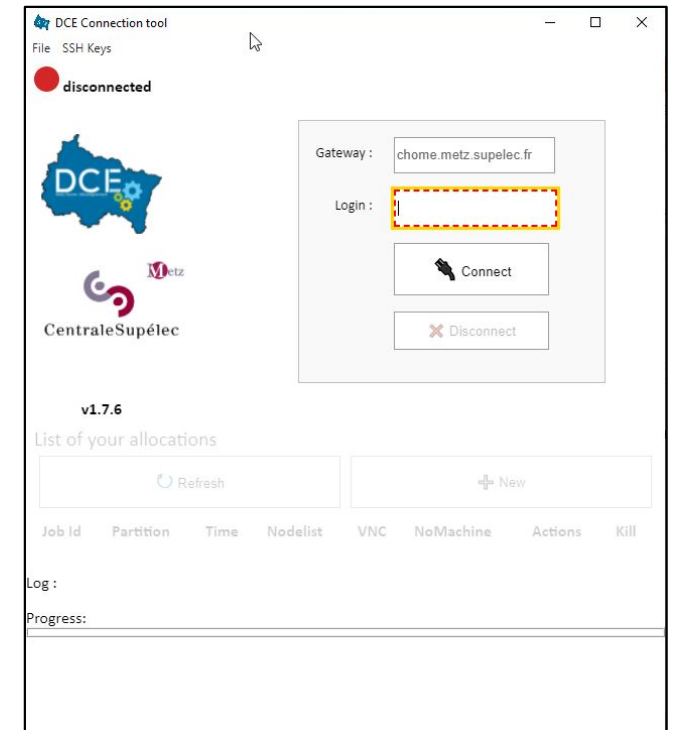
# CPU clusters

## CentraleSupélec Metz Campus



# CPU clusters

- Look at <https://tutos.metz.centralesupelec.fr>
- Look at **dcejs**
- Download & install the version for your system (Windows, Linux, Mac)



- Install a **VNC client/viewer**

Ex: TigerVNC on Windows/Mac



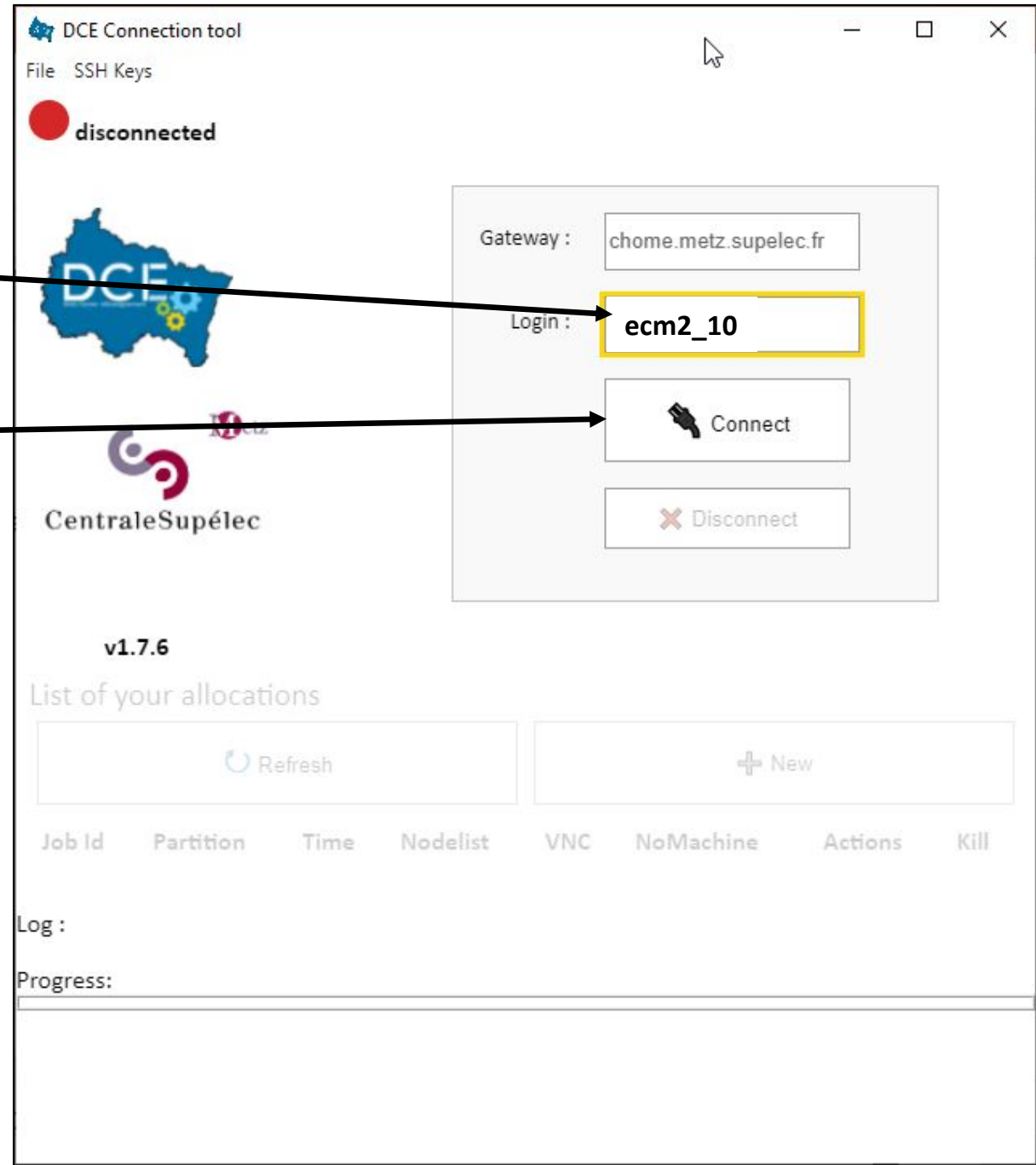
Download [vncviewer64-1.12.0.exe](https://sourceforge.net/projects/tigervnc/files/stable/1.12.0/) (or [vncviewer-1.12.0.exe](https://sourceforge.net/projects/tigervnc/files/stable/1.12.0/))

(on <https://sourceforge.net/projects/tigervnc/files/stable/1.12.0/>)

## CPU clusters

### 1. Launch dcejs

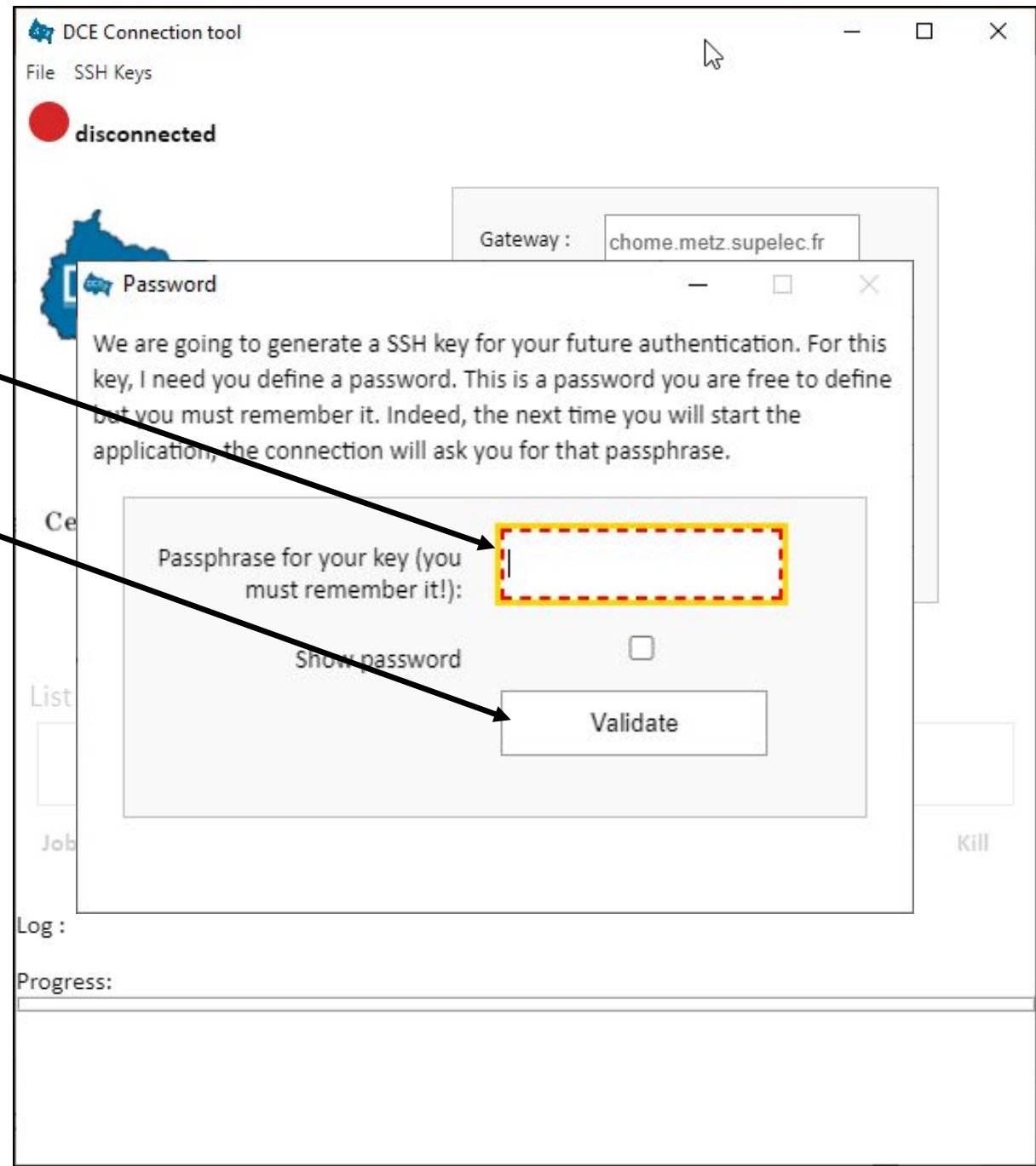
- Write your **ecmy\_xx** login
- Click on « Connect »



## CPU clusters

### 2. Launch dcejs

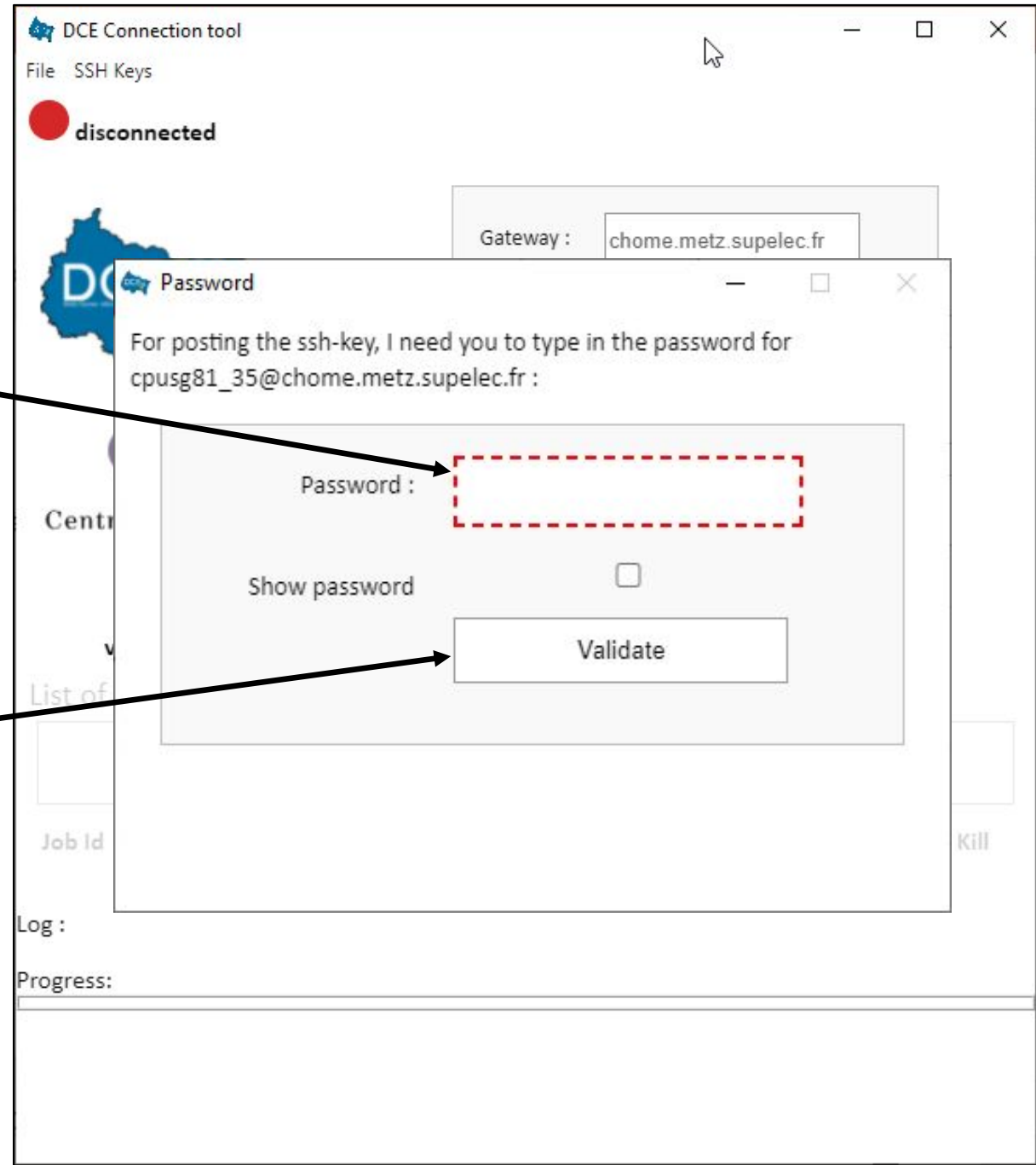
- .....
- Enter a passphrase  
(choose a basic one)
- Click on « Validate »



# CPU clusters

## 3. Launch dcejs

- .....
- During the first connexion to the DCE you will need to enter the passwd of your DCE login
- Click on « Validate »





# CPU clusters

## 4. Launch dcejs

- .....
- Click on « + New » to start to allocate some computing resources

The screenshot shows the 'DCE Connection tool' window. At the top, it indicates 'connected' with a green dot. Below this is the DCE logo and the CentraleSupélec logo. On the right, there are input fields for 'Gateway : chome.metz.supelec.fr' and 'Login : ecm2\_10', along with 'Connect' and 'Disconnect' buttons. Below the connection area, the version 'v1.7.6' is displayed. The 'List of your allocations' section contains a 'Refresh' button and a '+ New' button, which is highlighted by an arrow from the text in the previous slide. At the bottom, there is a table header with columns: Job Id, Partition, Time, Nodelist, VNC, NoMachine, Actions, and Kill. Below the table, there is a 'Log : Connected to the gateway.' message and a 'Progress:' indicator with a green bar.

# CPU c

## 5. Launch **dcejs** during the lab

- .....
- Click on « With res... »
- Enter a reservation code (*ask to the Prof*)
- Open settings
- **Uncheck the box**
- Enter: 2
- Enter: 1
- Click on « Validate »

The screenshot shows a 'New session' window with two tabs: 'Without reservation' and 'With reservation'. The 'With reservation' tab is active. Below the tabs, there is a text prompt: 'Please fill in the required parameters for you new allocation'. A 'Reservation' field is highlighted with a red dashed box, and a red error message below it reads 'The reservation code is required'. Below this, there is an 'Advanced settings' section with a dropdown arrow. Under 'Advanced settings', there is a checked checkbox labeled 'exclusive'. Below the checkbox are several input fields: '-c, --cpus-per-task', '-N, --nodes' (with the value '1' entered), '--qos', and '-n, --ntasks'. At the bottom, there is a 'Slurm command' field.

# CPU c

## 5. Launch **dcejs** after the lab

- .....
- Click on « **Without...** »
- Select partition **cpu\_tp**
- Specify **2:0** (2h)
- Open settings
- Uncheck the box
- Enter: 2
- Enter: 1
- Click on « **Validate** »

# CPU clusters

## 6. Launch dcejs

- .....
- Click on  
« Actions »
- Click on  
« Start VNC »

The screenshot shows the 'DCE Connection tool' window. A 'Session handler' sub-window is open, displaying three buttons: 'Start VNC', 'Start NoMachine', and 'Launch Xterm'. In the main window, there is a table with the following data:

Job Id	Partition	Time	Nodelist	VNC	NoMachine	Actions	Kill
3958	cpu_inter	0:04	kyle01	--	--	⊙Actions	Kill

Below the table, there is a log entry: 'Log : We are there, ready to work !' and a 'Progress:' indicator with a green bar.

# CPU clusters

## 7. Launch **dcejs**

- .....
- Get the local port number

Ex: 5916

- Launch your **VNC client** with all default options

Ex: TigerVNC



The screenshot shows the 'DCE Connection tool' window. At the top, it says 'connected' with a green dot. Below that is the DCE logo and the CentraleSupélec logo. On the right, there are input fields for 'Gateway : chome.metz.supelec.fr' and 'Login : ecm2\_10', along with 'Connect' and 'Disconnect' buttons. Below the logos, it says 'v1.7.6'. The main section is titled 'List of your allocations' and contains a table with columns: Job Id, Partition, Time, Nodelist, VNC, NoMachine, Actions, and Kill. The table has one row: Job Id 3959, Partition cpu\_inter, Time 0:23, Nodelist kyle01, VNC localhost:5916 (highlighted with a red box), NoMachine --, Actions (button), and Kill (button). Below the table, it says 'Log : VNC done. Please start your viewer.' and 'Progress:' with a green progress bar.

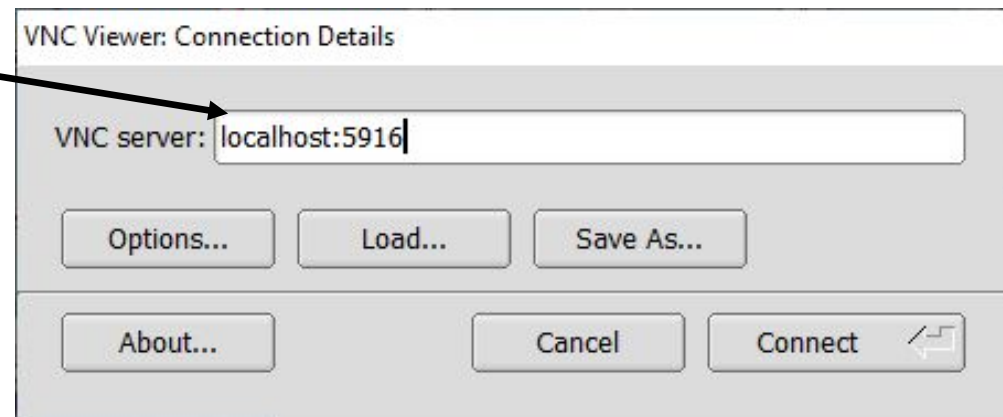
# CPU clusters

## 8. On windows:

- Launch your **VNC client** with all default options (ex: TigerVNC on Windows)



- Enter the port number returned by dcejs



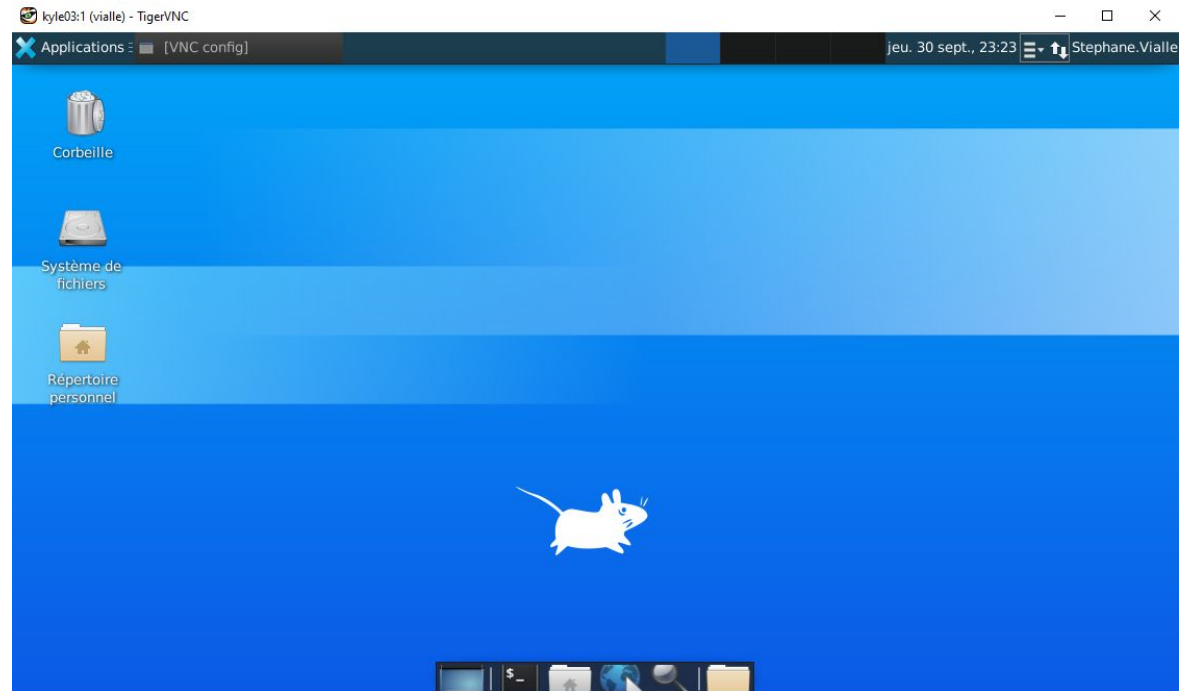
## 8. On Linux & Mac :

- It should be possible to just click on the port number in the dcejs window.

# CPU clusters

9. The desktop of the remote DCE machine appears

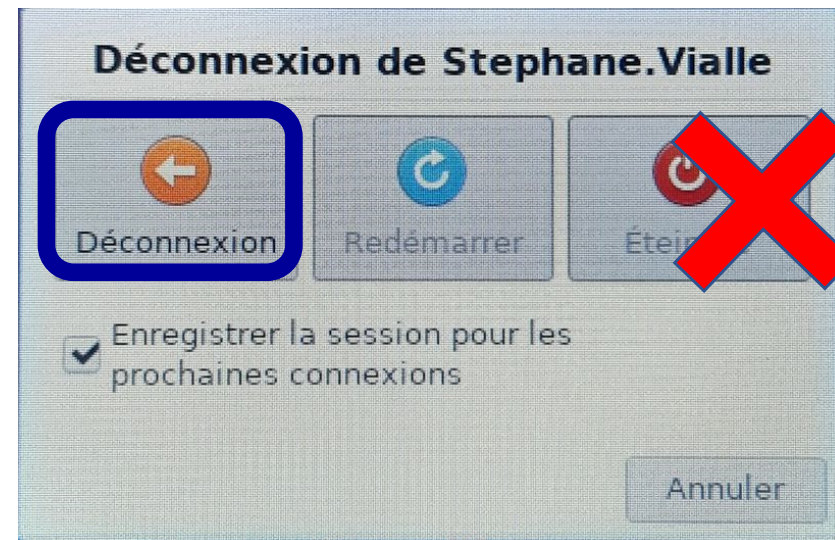
- You can launch a terminal, and an editor (code, xedit, ...)



10. When you disconnect:

**NEVER shut down the machine!**

Use the disconnect button

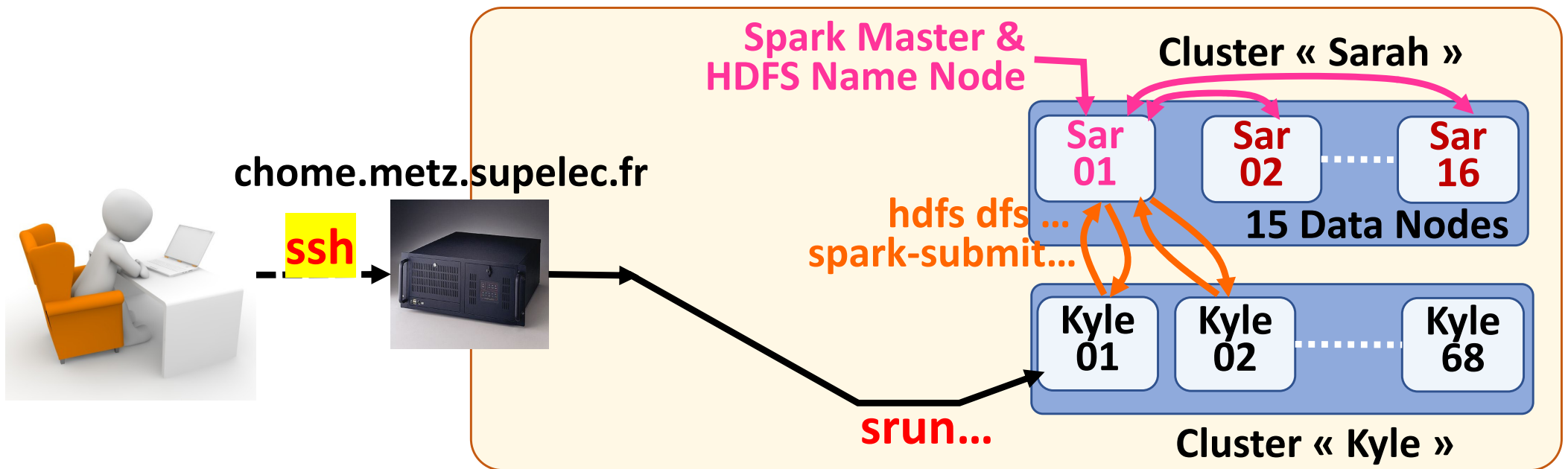


# Using Spark cluster of CentraleSupelec DCE

- DCE architecture and access with dcejs
- **DCE architecture and access with ssh**
- DCE access with vscode
- How to kill a zombie session



# Cluster connection commands

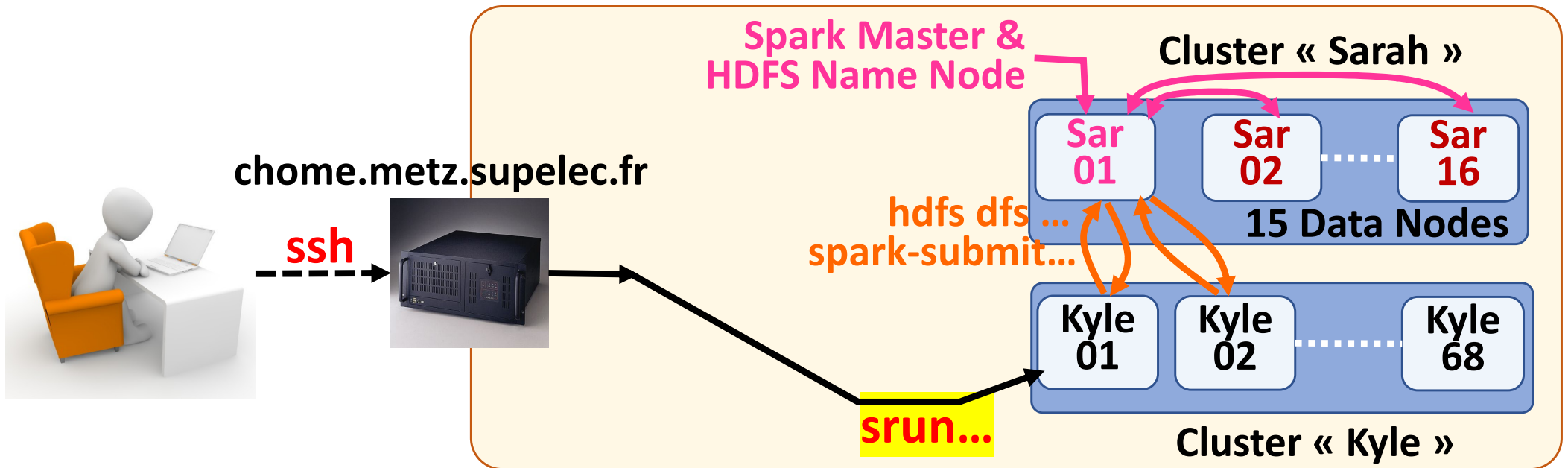


Linux → `ssh chome.metz.supelec.fr -l ecm2_1` *From ecm2\_1 up to ecm2\_15*

Windows → run a "powershell" and then the above command

- Old Windows →
- Download & Instal « putty »
  - « Session » menu : **phome.metz.supelec.fr**  
connection type : **ssh** (port 22)
  - « Connection » menu: set **Enable TCP keepalives**  
set **30s** between keepalives

# Cluster connection commands



On *chome* **DURING** the lab:

```
srun --reservation=myCode
```

Ask to the teacher

```
-N 1 --cpus-per-task=2
```

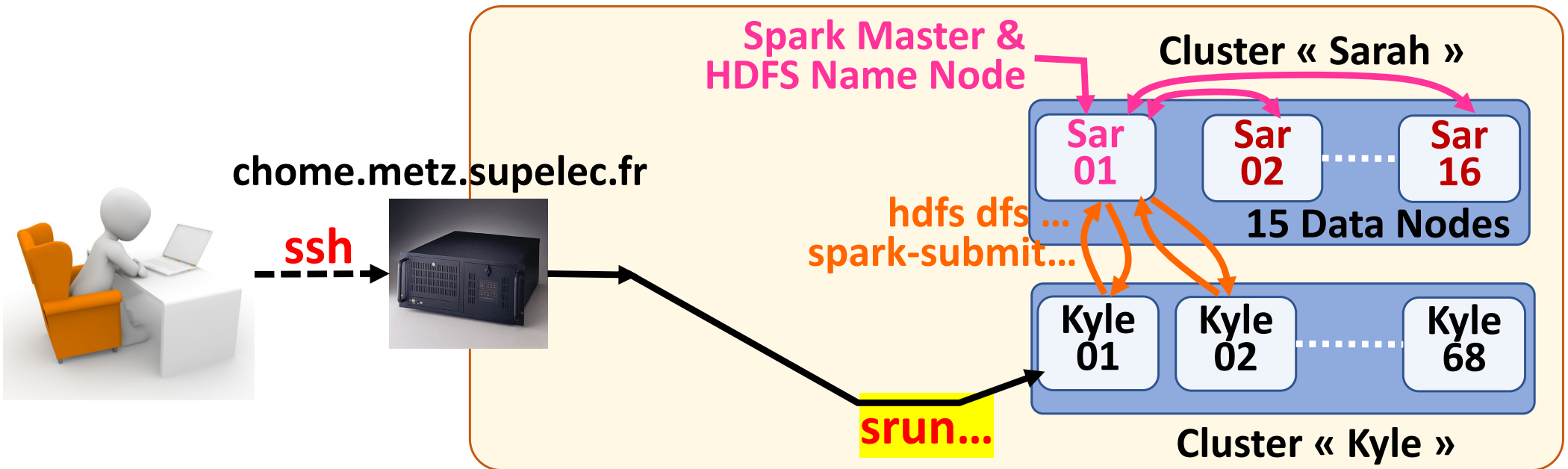
Allocate 2 logical  
cores on 1 node

```
--pty bash
```

Run an interactive session

**Write on ONE line!**

# Cluster connection commands



On *chome* **AFTER** the lab:

```

srun  -p cpu_tp
        -N 1 --cpus-per-task=2
        --pty bash
    
```

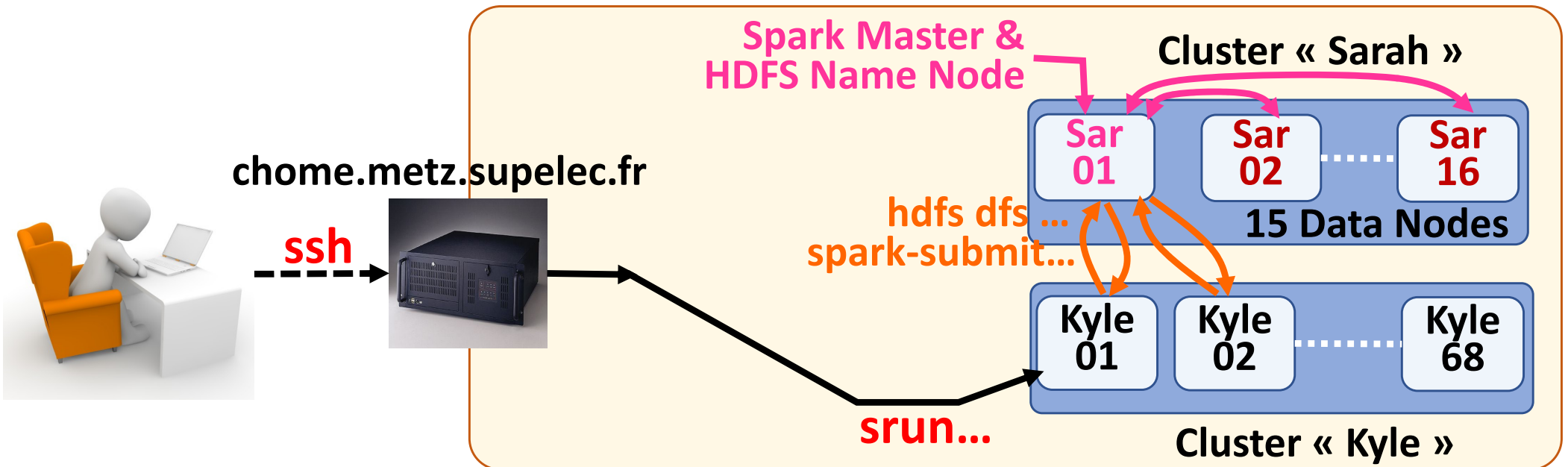
Partition to use

Allocate 2 logical cores on 1 node

Run an interactive session

**Write on ONE line!**

# Cluster connection commands



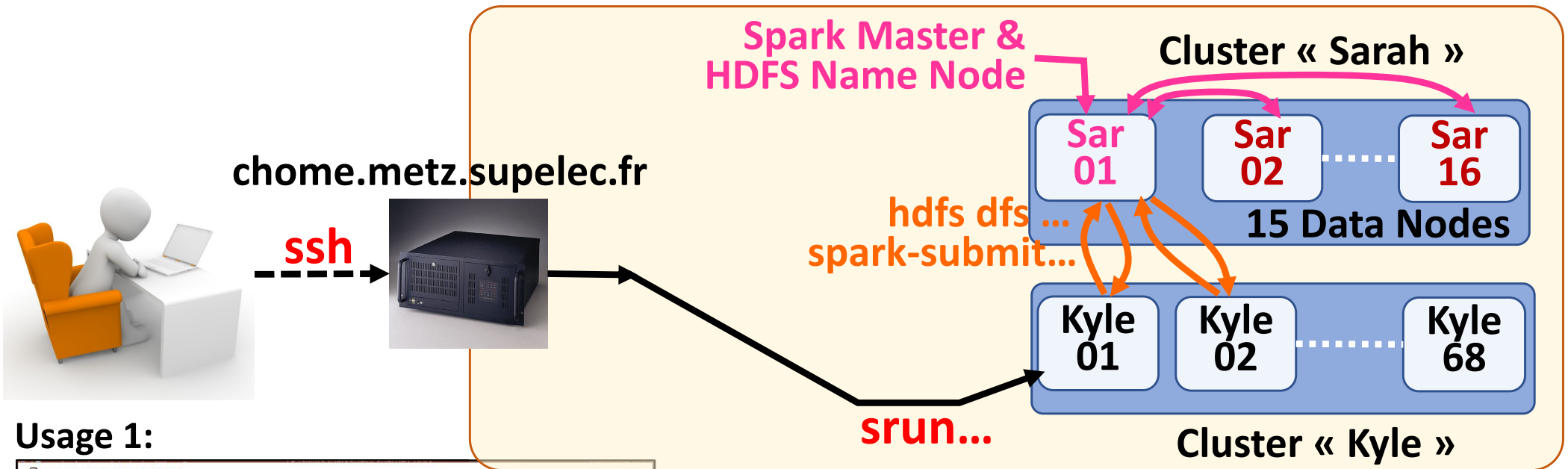
*On the cluster node:*

`sinfo -l` → information on partitions

`squeue` → information on job queues

`scontrol show job` → information on running jobs

# Edition of remote files



## Usage 1:

```

phome.metz.supelec.fr - PuTTY
GNU nano 2.9.3          template.py
import sys
import os
from pyspark import SparkContext, SparkConf

# MapReduce code (TO DEVELOP) -----
def avg_temp_NN(theText_file):
    temperatures = theText_file.map(lambda line: line.split(","))\
    . . . . .
    return temperatures

# Main code -----
# - generate the input and output file names (TO COMPLETE)
file_name = os.path.basename(sys.argv[1])
input_file = "hdfs://sar01:9000/data/temperatures/" + file_name
output_file = "hdfs://sar01:9000/cpucsXX/cpucsXX_YY/" + os.path.sp$
# - create the Spark context
sc = SparkContext()

[ Lecture de 23 lignes (converties du format DOS) ]
Aide  Écrire  Chercher  Couper  Justifier
Quitter Lire fich. Remplacer Coller  Analyse stat
    
```

On the cluster node

Python-Spark file: remote edition (ex: vi / nano editor)

```

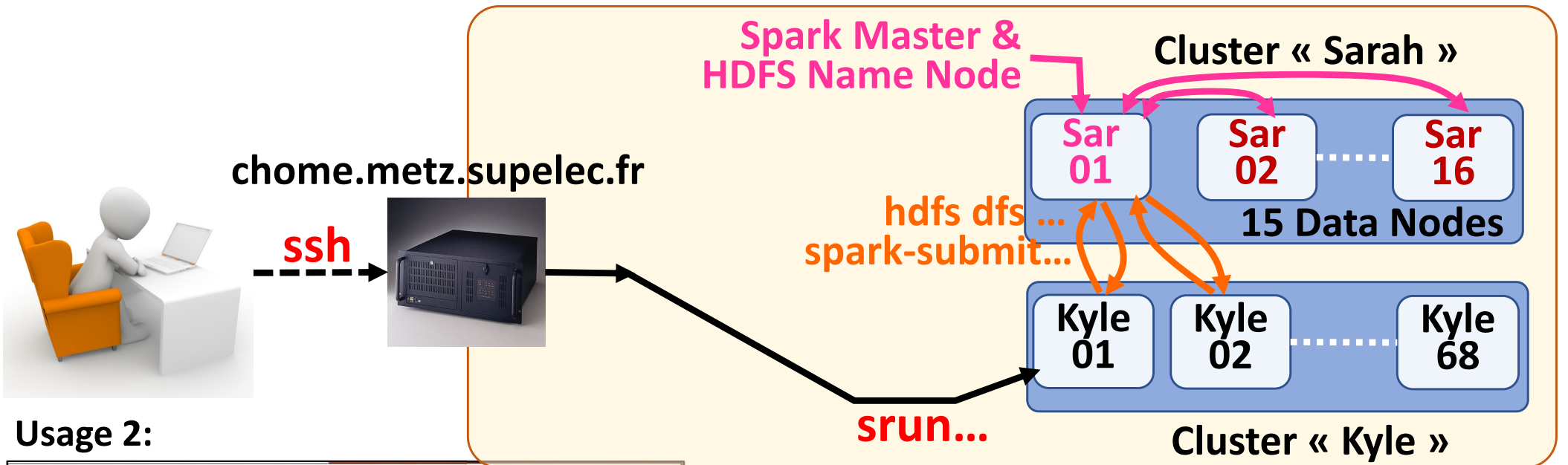
phome.metz.supelec.fr - PuTTY
cpucs12_10@kyle18:~$ spark-submit --master spark://sar01:7077
template.py input.csv
    
```

Spark-submit

On the cluster node

Distributed spark execution (remote spark-submit command)

# Transfer of remote files



## Usage 2:

```

template.py [C:\Users\vialle\Desktop\port-enseignement\SupportEnseignement\CS-C...
File Edit View Settings ?
Find Replace Word W
1 import sys
2 import os
3 from pyspark import SparkContext, SparkConf
4
5 # MapReduce code (TO DEVELOP) -----
6 def avg_temp_NN(theText_file):
7     temperatures = theText_file.map(lambda line:
8     .....
9     .....
10    .....
11    # Main code -----
12    # -- generate the input and output file names (T
13    file_name = os.path.basename(sys.argv[1])
14    input_file = "hdfs://sar01:9000/data/temperatur
15    output_file = "hdfs://sar01:9000/cpucsxx/cpucsx
16    # -- create the Spark context
17    sc = SparkContext()
Ln 1: 24 Col 1 Sel 0 891 octet(s) UTF-8 CR
    
```

Python Spark file:  
local edition

Desktop - phome - WinSCP

Local Marquer Fichiers Commandes Session Options Distant Aide

phome X Nouvelle session

C:\Users\vialle\Desktop\ /usr/users/ims/vialle/DCE-Spark/

chome.metz.supelec.fr

File transfer (winSCP)

```

phome.metz.supelec.fr - PuTTY
cpucs12_10@kyle18:~$ spark-submit --master
spark://sar01:7077 template.py input.csv
    
```

On the cluster node

Remote distributed  
spark execution  
(spark-submit cmd)

# Using Spark cluster of CentraleSupelec DCE

- DCE architecture and access with dcejs
- DCE architecture and access with ssh
- **DCE access with vscode**
- How to kill a zombie session

# Edition of remote files

**Configuration and usage of vscode to reach the DCE:**

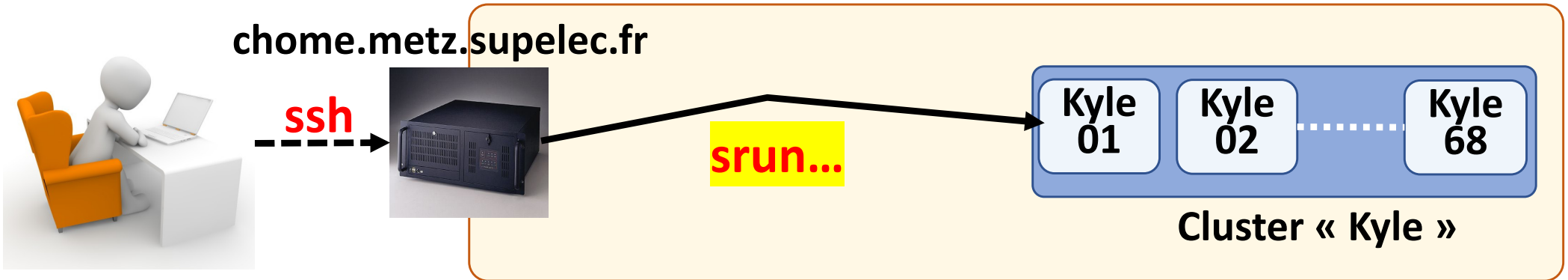
<https://webtv.centralesupelec.fr/videos/how-to-connect-to-dce-with-visual-studio-code/>



# Using Spark cluster of CentraleSupelec DCE

- DCE architecture and access with dcejs
- DCE architecture and access with ssh
- DCE access with vscode
- **How to kill a zombie session**

# DCE connection commands



On the *cluster node* or on chome:

**mysrun** → information on your running *srun --pty bash*

Ex: `cpu_vialle@chome:~$ mysrun`

```
506 bash RUNNING 11:08 4:30:00 1 kyle01 k1
```

**scancel <jobId>** → delete a *slurm job*

Ex: `cpu_vialle@chome:~$ scancel 506`

To remove a zombie  
job blocking new  
resource allocations!

Big Data – TP1 Part 0

# Connection to the DCE of CentraleSupélec DCE using *dcejs* or *ssh* or *vscode*

(Data Center for Education)

**Questions ?**