

ARC Cvičení 1

Práce se superpočítačem, užitečné aplikace

Marta Jaroš

Supercomputing Technologies Research Group
Brno University of Technology, Faculty of Information Technology
Božetěchova 1/2, 612 66 Brno - Královo Pole
icudova@fit.vutbr.cz

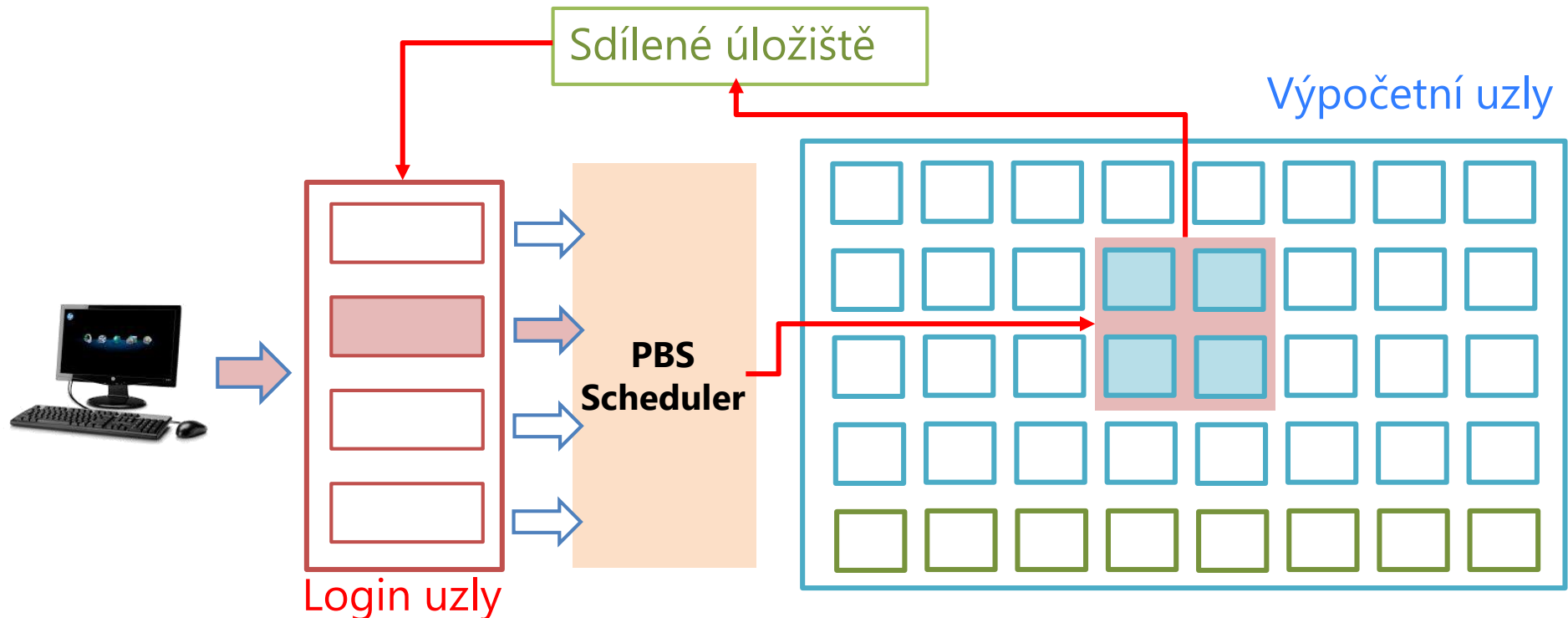


- Jak se připojit na cluster
- Připojení disku
- Moduly, jobscripty
- Odeslání jobu, smazání, monitorování
- VNC (virtual network computing)
- Intel tools a Allinea (ARM) tools
- Příklad (matrix multiplication, hello world)

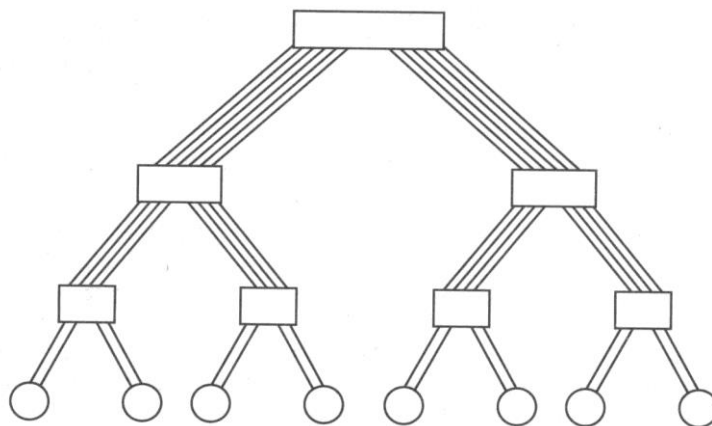
IT4Innovations
national
supercomputing
center



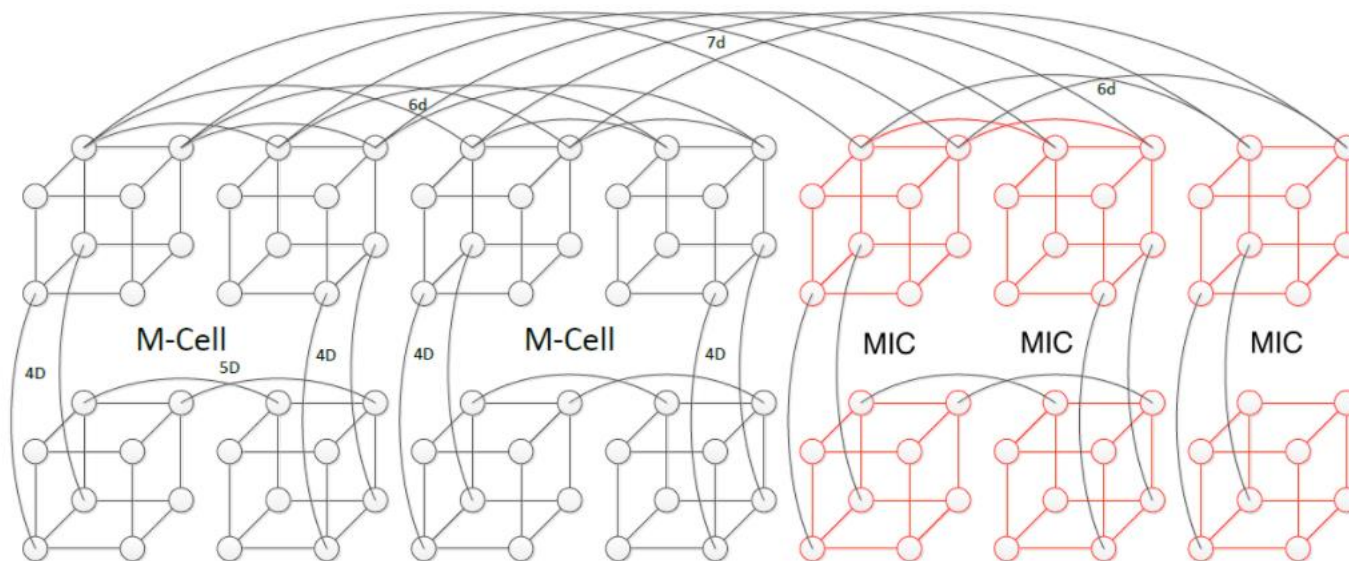
- **Login** uzly a **výpočetní** uzly (mohou obsahovat akcelerátory, např. GPU)
- **Fronty**
 - **qprod** – projekty
 - **Rezervace** (např. R1234567) – na cvičeních
 - qexp, qfree, qnvidia, qmic, ...



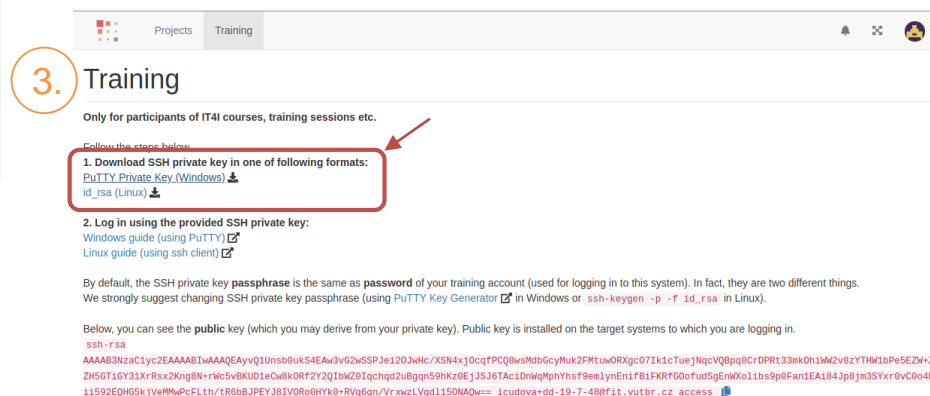
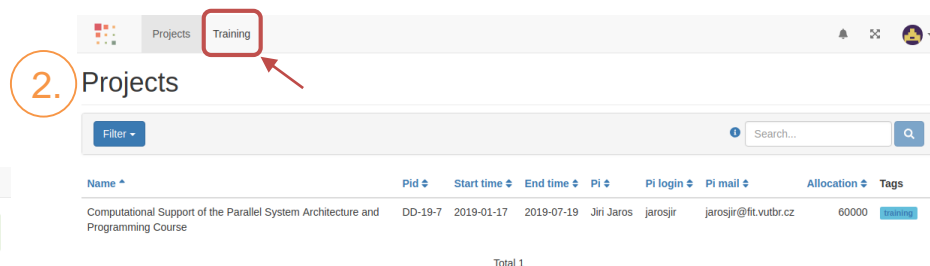
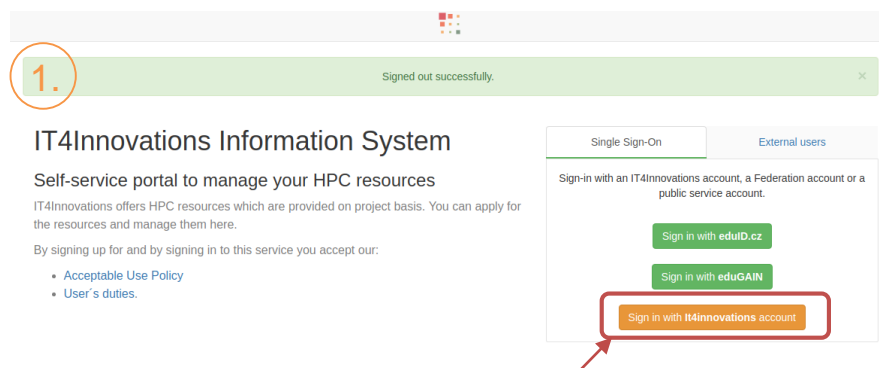
- Celkem 209 výpočetních uzlů, 3344 výpočetních jader
 - **180 uzlů obsahuje 2x8 jádro Sandy Bridge-E, min 64 GB RAM na uzel**
 - **23 akcelerovaných uzlů (NVIDIA K20)**
 - 4 akcelerované uzly (Intel Xeon Phi)
 - 2 tlusté uzly – 512 GB RAM, 2x100 GB SSD
- Celkem 15TB RAM a 94TFLOPs
- Propojení 40Gb/s **Infiniband**, neblokující tlustý strom



- **1008 výpočetních uzlů** (576 bez akcelérátoru)
 - Každý uzel **2x12 jádro Intel Xeon E5, 128 GB RAM**
 - Z toho **432 akcelerovaných uzlů (Intel Xeon Phi)**
- Propojení 56Gb/s **Infiniband**, 7D-hyperkostka



- Dostali jste ústřížek se svým **loginem** (**dd-xx-yy-zz**) a **heslem** (lze jej změnit – některé znaky mohou dělat problém v bashi).
- Přihlaste se s tímto loginem a heslem do **extranet.it4i.cz**
- **Stáhněte si klíč** (soubor id_rsa)



Mohlo by se hodit:

- IT4i dokumentace (Linux, Windows-Putty)
- MobaXterm (alternativa k Putty)

FYI: Zde se nacházíte.

Složky, ...

Nastavte *read-write* práva JEN pro vás.

Připojte se ke clusteru pomocí hesla.

```
Terminal
local $ ssh -i /path/to/id_rsa username@cluster.it4i.cz

local $ pwd
/home/xlogin
local $ ls -a
Desktop      Downloads
Documents    Music
.ssh
...

local $ chmod 600 ~/.ssh/id_rsa
local $ ssh -i .ssh/id_rsa dd-xx-yy-zz@salomon.it4i.cz
```

Ctrl+H k zobrazení skrytých souborů.

Zde si uložte svůj id_rsa klíč.

Pouze jednou po stažení klíče.

You will be asked to write the key passphrase.

Salomon

<http://www.it4i.cz/?lang=en>

dd-xx-yy-zz@salomon.it4i.cz

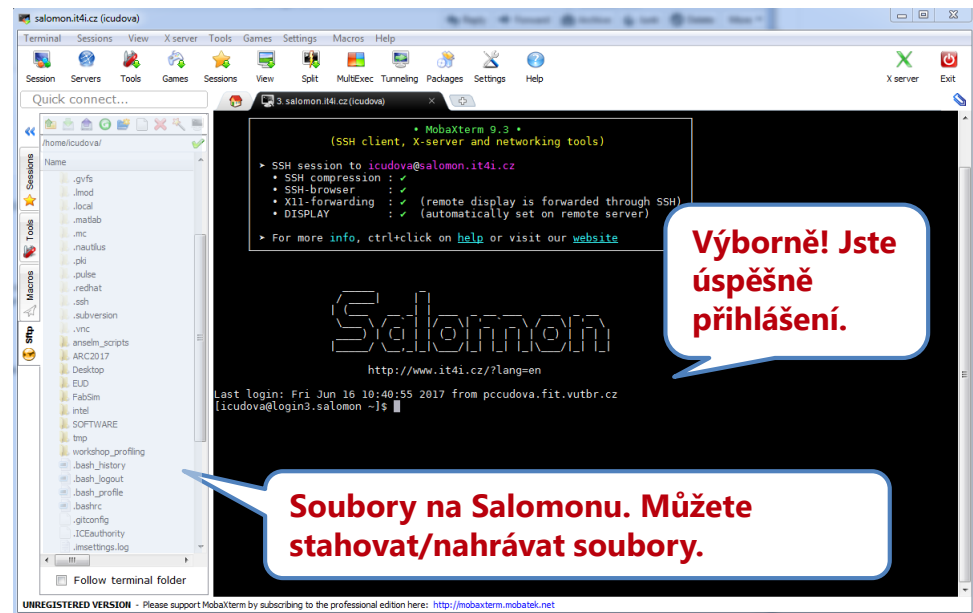
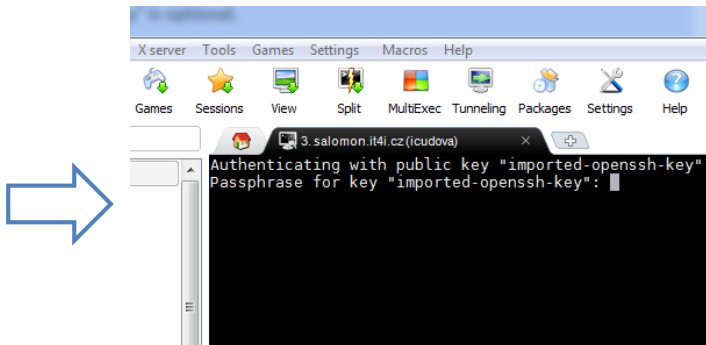
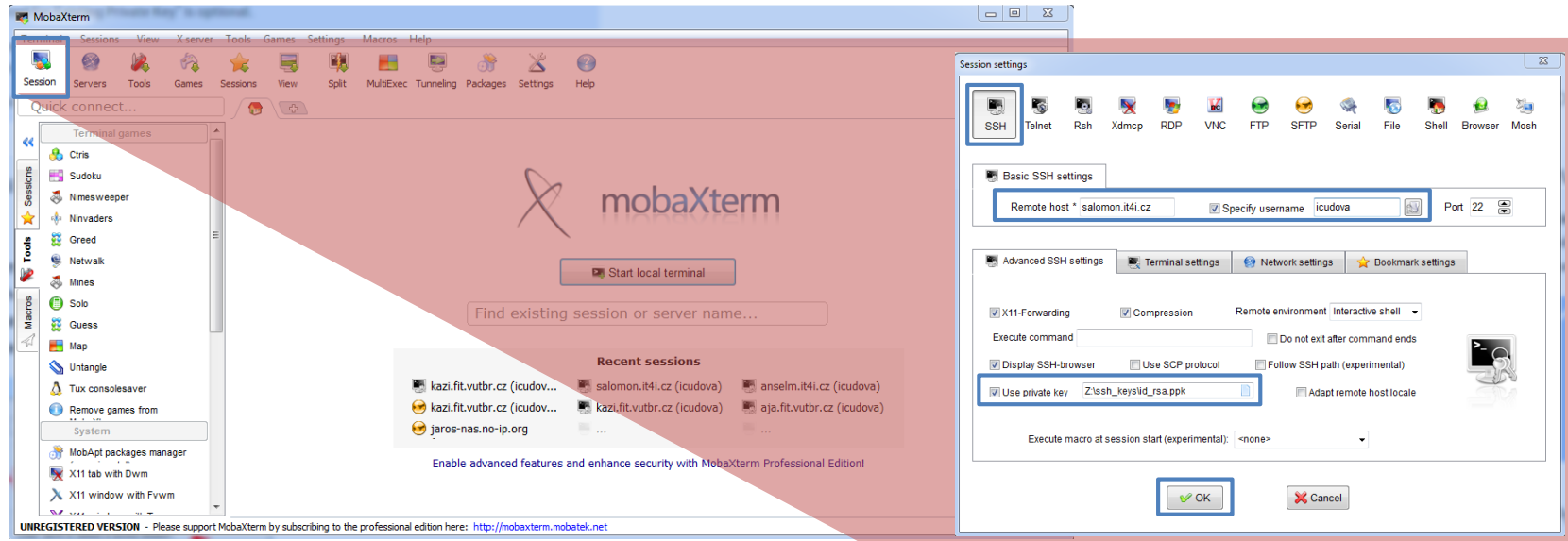
nebo

dd-xx-yy-zz@anselm.it4i.cz

```
[dd-xx-yy-zz@login2.salomon ~]$
```

Výborně! Jste úspěšně přihlášení.

Připojení se ke clusteru (Windows, .ppk klíč)



Vytvořte prázdnou složku k připojení disku Salomonu.

FYI: Běžte do lokální složky *salomon-disk* a zobrazte obsah.

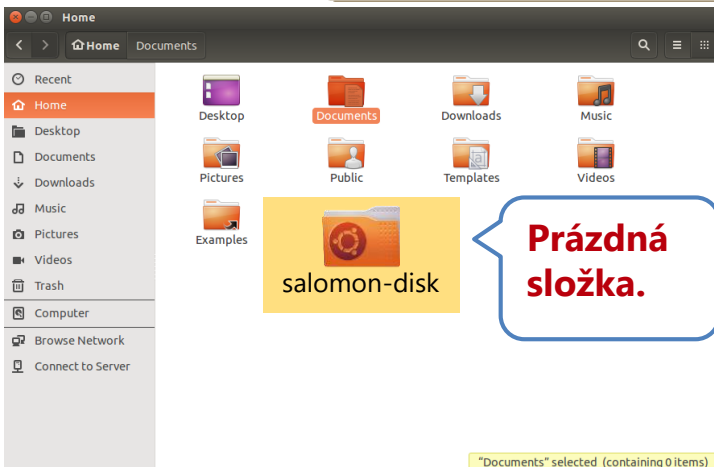
```
Terminal
me@linuxbox:~$ local $ sshfs -o IdentityFile=/path/to/id_rsa
dd-xx-yy-zz@cluster.it4i.cz:~. Mountpoint

local $ mkdir salomon-disk
local $ sshfs -o IdentityFile=/home/xlogin/.ssh/id_rsa
dd-xx-yy-zz@salomon.it4i.cz:~. salomon-disk
You will be asked to write the key passphrase.

local $ cd salomon-disk; ls -la
Documents      .
Projects       .ssh
```

Úplná cesta ke klíči!!

Složky na Salomonu.



sshfs ...



Host salomon

```
HostName    salomon.it4i.cz
User        xlogin
IdentityFile ~/.ssh/id_rsa
```

1. Vytvoření **config souboru** ve složce .ssh.



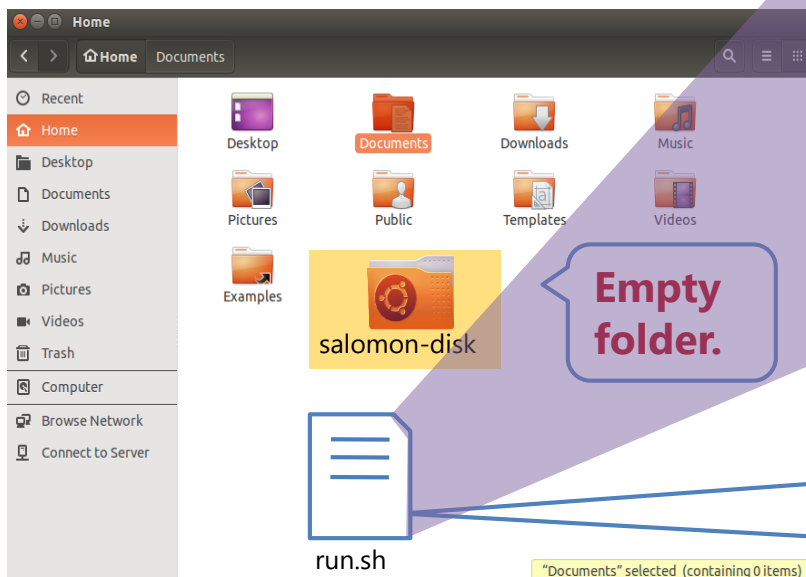
config

Přístup ke clusteru
jednoduše
ssh salomon

2. Zkopírujte lokální **veřejný klíč** do
souboru **authorized_keys** na
clusteru (složka .ssh).

-> přístup bez zadávání hesla

```
#!/bin/bash
case "$1" in
'salomon')
echo "Mounting salomon..."
sshfs salomon:/home/username
~/Documents/salomon-disk
echo "done" ;;
esac
```



Empty
folder.



run.sh

"Documents" selected (containing 0 items)

3. Vytvořte **bash script** k připojení
vzdálených disků. K připojení disku na
Salomonu ho spusťte jako
bash run.sh salomon

- Překladače, knihovny, debuggery a profilery, vývojářské nástroje a jiný software...
 - Nastavení proměnných prostředí a cest...
- => Způsob jak nastavit celé prostředí jedním příkazem

- Vylistování modulů
- Načtení modulu
- Odstranění modulu
- Odstranění všech modulů
- Zobraz načtené moduly

```
$ module avail **or** ml av
```

```
$ module load intel **or**  
$ ml intel
```

```
$ module unload intel **or**  
$ ml -intel
```

```
$ module purge
```

```
$ module list **or** ml
```

1. Načtěte modul GCC (lze doplňovat pomocí tab).
 - Jakou má verzi?
 - Jaké verze jsou dostupné?
2. Vylistujte si načtené moduly.
3. Odstraňte modul GCC.

- **Odeslání jobu** – pošli job do fronty (k výpočtu)
 - **Interactive job** – super pro vývoj a debuggování

Salomon – 24 jader
Anselm – 16 jader

```
$ qsub -A DD-XX-YY -q qprod -l select=5:ncpus=24 -I -X
```

- **Jobscript** – automatizace výpočtu

```
$ qsub -A DD-XX-YY -q qprod -l select=5:ncpus=24,walltime=08:00:00  
./myjob **or** see next slide
```

- **Monitorování jobu** – jaké joby jsou ve frontě, které běží a které už doběhly.

```
$ qstat -u $USER  
srv11:  
Job ID Username Queue Jobname SessID NDS TSK Memory Time S Time  
16287.isrv5 icudova qprod job1 6183 1 24 -- 2:0 R 00:25  
16468.isrv5 icudova qprod job2 8060 4 96 -- 48:0 Q --:--  
$ qstat -f 16287.isrv5 # to see complete job details
```

- **Smazání jobu** – (zastav výpočet a) vymaž job z fronty.

```
$ qdel 16287.isrv5
```

```
#!/bin/bash
#PBS -q qprod          # queue name
#PBS -N mult           # name of your job
# give me a whole node for 8 hours (max)
#PBS -l select=1:ncpus=24,walltime=8:00:00
#PBS -A DD-19-7        # project ID
#PBS -o ../outlog      # folders for err- and outlog
#PBS -e ../errlog

# start job from the directory it was submitted
cd $PBS_O_WORKDIR

# load modules
ml intel

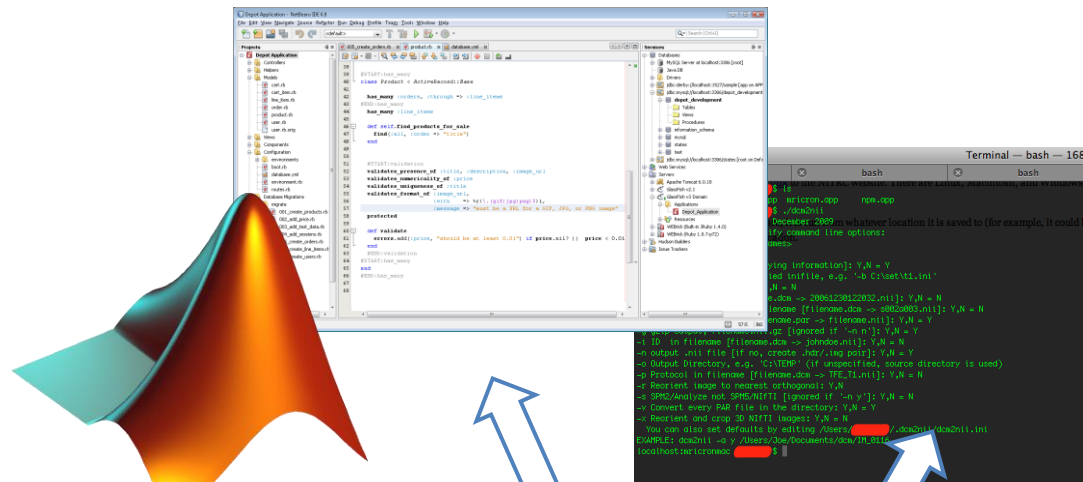
# run your code
./mult 54 78
```

Do fronty odešli job jako
`qsub job.pbs`



Your PC ran into a problem that it couldn't handle, and now it needs to restart.

You can search for the error online: HAL_INITIALIZATION_FAILED



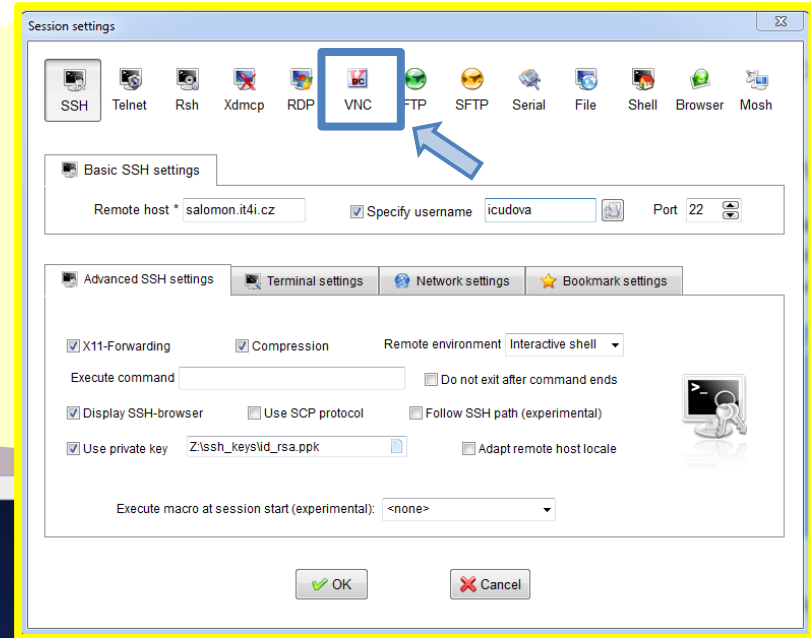
Váš počítač

vs.



```
Terminal
me@linuxbox:~$
```

\$ vncviewer
127.0.0.1:5913

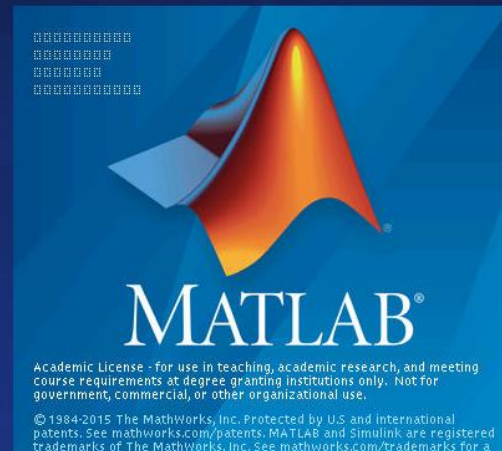


```
icudova@r31u11n844:~
```

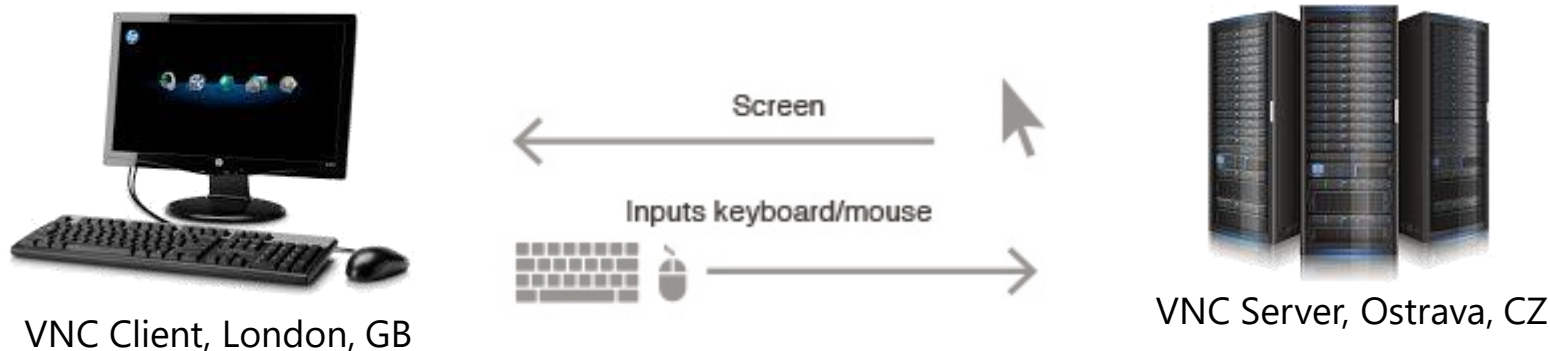
File Edit View Search Terminal Help

```
[icudova@login4.salomon Tests]$ xhost +
access control disabled. Any host can connect.
[icudova@login4.salomon Tests]$ qsub -I -v DISPLAY=$(uname -n):$(echo $DISPLAY |
cut -d ':' -f 2) -A OPEN-9-12 -q qprod -l select=1:ncpus=24 -l walltime=8:0:0
qsub: waiting for job 1130330.isrv5 to start
qsub: job 1130330.isrv5 ready

[icudova@r31u11n844 ~]$ ml MATLAB
[icudova@r31u11n844 ~]$ matlab &
[1] 11499
[icudova@r31u11n844 ~]$ MATLAB is selecting SOFTWARE OPENGGL rendering.
```



- Vzdálené připojení ke GUI přes síť
- **Rychlejší** než X11 (**ssh -X ...**)
- Zavřením okna s VNC clientem **se vzdálené výpočty nezastaví.**
- Připojte se k login uzlu a pak si vemte výpočetní uzel k dosažení maximálního výkonu – může být rovněž použito ke spuštění aplikací GUI.



- **Přihlášení se k Anselmu.**
 - Vytvoření VNC hesla.
 - Start VNC serveru.
 - Výběr „display number“.
 - Běží už VNC server?
- Běžte zpět ke svému počítači
 - Vytvoření tunelu
 - Test, zda je tunel vytvořen
 - Připojení se s VNC klientem
- Ukončení práce

```
local$ ssh -i .ssh/id_rsa  
dd-xx-yy-zz@anselm.it4i.cz
```



**VNC
dokumentace pro
Anselm/Salomon**

- Přihlášení se k Anselmu.
 - **Vytvoření VNC hesla.**
 - Start VNC serveru.
 - Výběr „display number“.
 - Běží už VNC server?
- Běžte zpět ke svému počítači
 - Vytvoření tunelu
 - Test, zda je tunel vytvořen
 - Připojení se s VNC clientem
- Ukončení práce

```
[dd-??@login2 ~]$ vncpasswd  
Password:  
Verify:
```

Local VNC heslo by mělo být vytvoření před **prvním** přihlášením.

- Přihlášení se k Anselmu.
 - Vytvoření VNC hesla.
 - **Start VNC serveru.**
 - **Výběr „display number“.**
 - Běží už VNC server?
- Běžte zpět ke svému počítači
 - Vytvoření tunelu
 - Test, zda je tunel vytvořen
 - Připojení se s VNC clientem
- Ukončení práce

Procesy uživatelů.

```
[dd-??@login2 ~]$ ps aux | grep Xvnc
username 5971 ... /usr/bin/Xvnc :79 ...
username 6029 ... /usr/bin/Xvnc :60 ...
```

Již použitá „display numbers“ na **login 2**.

Vyberme si třeba číslo 13.

```
[dd-??@login2 ~]$ vncserver :13
                        -geometry 1600x900 -depth 16
```

Nastavení zobrazení.

```
New 'login2:1 (username)' desktop is
login2:1
Starting applications specified in
/home/dd-??/.vnc/xstartup
Log file is
/home/dd-??/.vnc/login2:1.log
```

- Přihlášení se k Anselmu.
 - Vytvoření VNC hesla.
 - **Start VNC serveru.**
 - Výběr „display number“.
 - **Běží už VNC server?**
- Běžte zpět ke svému počítači
 - Vytvoření tunelu
 - Test, zda je tunel vytvořen
 - Připojení se s VNC clientem
- Ukončení práce

```
[dd-??@login2 ~]$ ps aux | grep Xvnc  
dd-??  5978  ... /usr/bin/Xvnc :13 ...  
username 6029  ... /usr/bin/Xvnc :60 ...
```

nebo ...

```
[dd-??@login2 ~]$ vncserver -list  
TigerVNC server sessions:
```


X DISPLAY #	PROCESS ID
:13	18437

- Přihlášení se k Anselmu.
 - Vytvoření VNC hesla.
 - Start VNC serveru.
 - Výběr „display number“.
 - Běží už VNC server?
- **Běžte zpět ke svému počítači**
 - **Vytvoření tunelu**
 - **Test, zda je tunel vytvořen**
 - Připojení se s VNC clientem
- Ukončení práce

- Otevřete nový terminál na vašem počítači (Ctrl+Alt+T).
- Tunel se musí odkazovat na **stejné číslo login uzlu**, odkud byl VNC server spuštěn (login2 v tomto případě).

```
ssh -TN -f username@loginX.cluster-  
name.it4i.cz -L  
5900+DN:localhost:5900+DN
```

Šablona



```
local $ ssh -TN -f  
dd-??@login2.salomon.it4i.cz -L  
5913:localhost:5913
```

vncserver běží na portu **5900** + **display number**.

```
local $ netstat -natp | grep 5913  
tcp 0 0 127.0.0.1:5913 0.0.0.0:* ...  
tcp6 0 0 :::5913 :::* LISTEN 2022/ssh
```

SSH tunel běží na pozadí.

Po dokončení práce zničte tento tunel pomocí příkazu: **kill 2022**

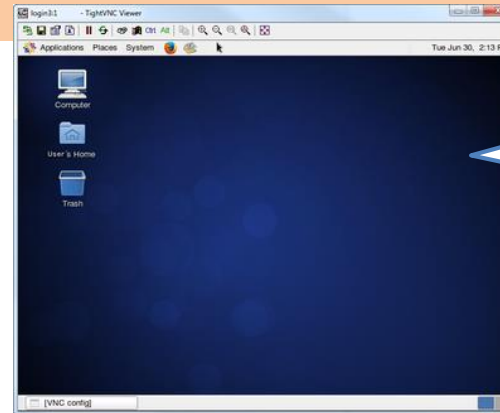
```
local $ export no_proxy="localhost"
```

Konfigurace počítačové stě v **CVT** vyžaduje **zakázání proxy** pro lokální adresy, aby se VNC mohlo připojit.

- Přihlášení se k Anselmu.
 - Vytvoření VNC hesla.
 - Start VNC serveru.
 - Výběr „display number“.
 - Běží už VNC server?
- Běžte zpět ke svému počítači
 - Vytvoření tunelu
 - Test, zda je tunel vytvořen
 - **Připojení se s VNC klientem**
- Ukončení práce

```
local $ vncviewer 127.0.0.1:5913
```

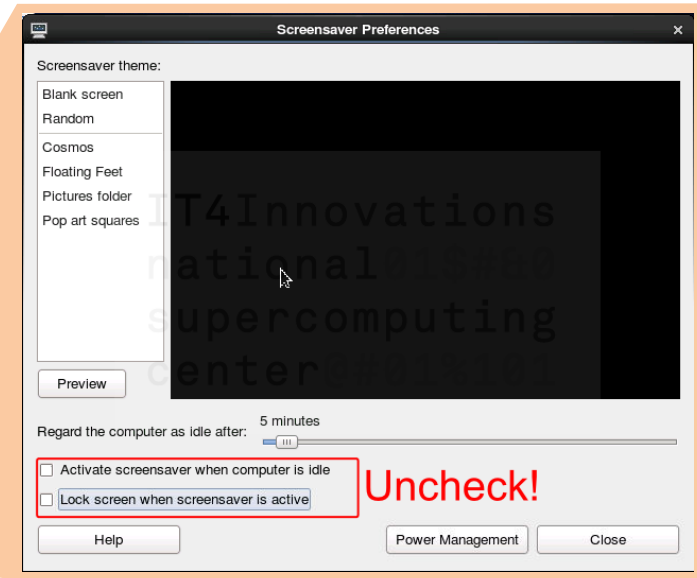
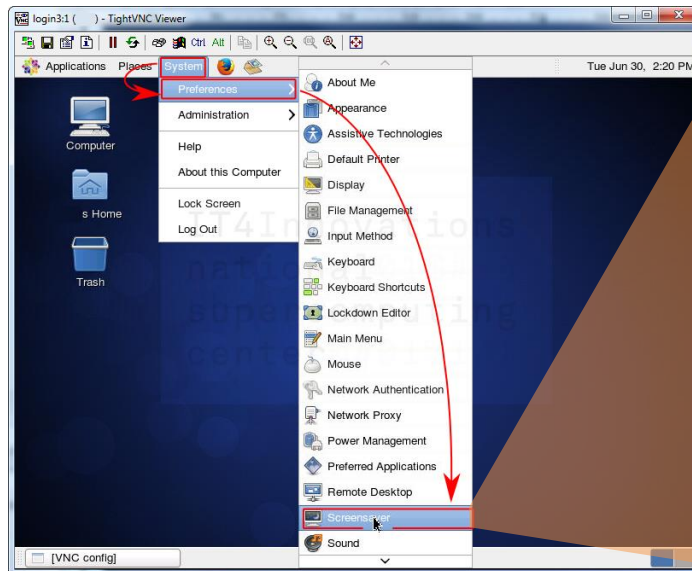
Bude vyzváni k zadání VNC hesla.



Úspěšné přihlášení.

VNC klienta najdete ke stažení ve wisu u cvičení.

Malý hack se spořičem obrazovky.



- Přihlášení se k Anselmu.
 - Vytvoření VNC hesla.
 - Start VNC serveru.
 - Výběr „display number“.
 - Běží už VNC server?
- Běžte zpět ke svému počítači
 - Vytvoření tunelu
 - Test, zda je tunel vytvořen
 - Připojení se s VNC clientem
- **Ukončení práce**

```
[dd-??@login2 ~]$ vncserver -kill :13  
nebo
```

```
[dd-??@login2 ~]$ pkill vnc
```

```
** Zničí VNC Server na vzdáleném  
výpočetním prostředku. **
```

```
local $ kill 2022
```

```
** Zničí SSH tunel na lokálním  
počítači. **
```


- **Arm (Allinea) Performance Reports**

```
ml PerformanceReports  
perf-report ./prog [parameters]
```

- Informace o výpočtu, komunikaci a IO
- Generuje jednoduché a přehledné .txt a .html

- **Arm (Allinea) MAP**

```
ml Forge  
map ./prog [parameters]
```

- Profiler (ukáže řádky aplikace, kde to „drhne“)
- Jednovláknové programy, OpenMP, MPI

- **Arm (Allinea) DDT**

```
ml Forge  
ddt ./prog [parameters]
```

- Grafický debugger
- Podpora jedno- a více vláknových programů, OpenMP, MPI, CUDA, hybridní kódy
- Podpora jazyků C, C++, Fortran

- Nástroje vyžadují překlad s **ladícími symboly**
- Vždy je jednodušší krokovat menší problémy

- Intel Inspector

```
ml inspector_xe (Anselm) | ml Inspector (Salomon)
    inspxe-gui
```

- Debuggovací nástroj, který detekuje memory leaks, race conditions, neinicializované proměnné, ...

- Intel Advisor

```
ml Advisor
    advixe-gui
```

- Ukazatel základních metrik (vektorizace, počet běžících vláken)
- Roofline analýza

- Intel Vtune Amplifier

```
$ qsub ... -l vtune=version_string (př. 2017_update2)
    ml Vtune
    amplxe-gui
```

- Detailní analýza kódu, používá HW counters
- Např. monitoruje výpadky v cache, synchronizace, I/O, ...

Na Salomonu jsou nejnovější verze těchto aplikací.