CFD on HPC – OpenFOAM example

Using HPCFS

Setting up the NoMachine client

Available for installation at page https://www.nomachine.com/download-enterprise#NoMachine-Enterprise-Client

I. *Select New*

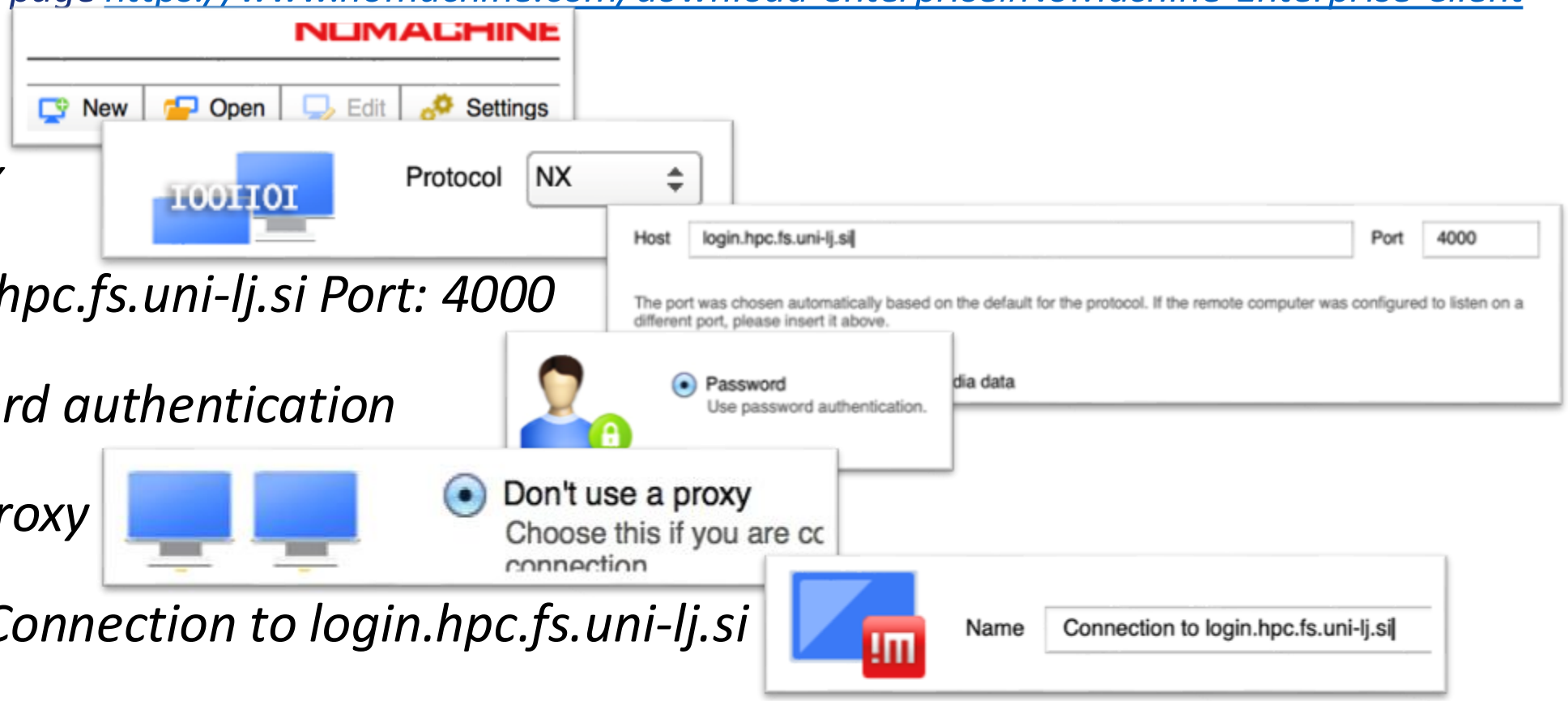
II. *Protocol NX*

III. *Host: login.hpc.fs.uni-lj.si Port: 4000*

IV. *Use Password authentication*

V. *Don't use proxy*

VI. *Done with Connection to login.hpc.fs.uni-lj.si*



Connecting to HPCFS

I. *Select and Connect*

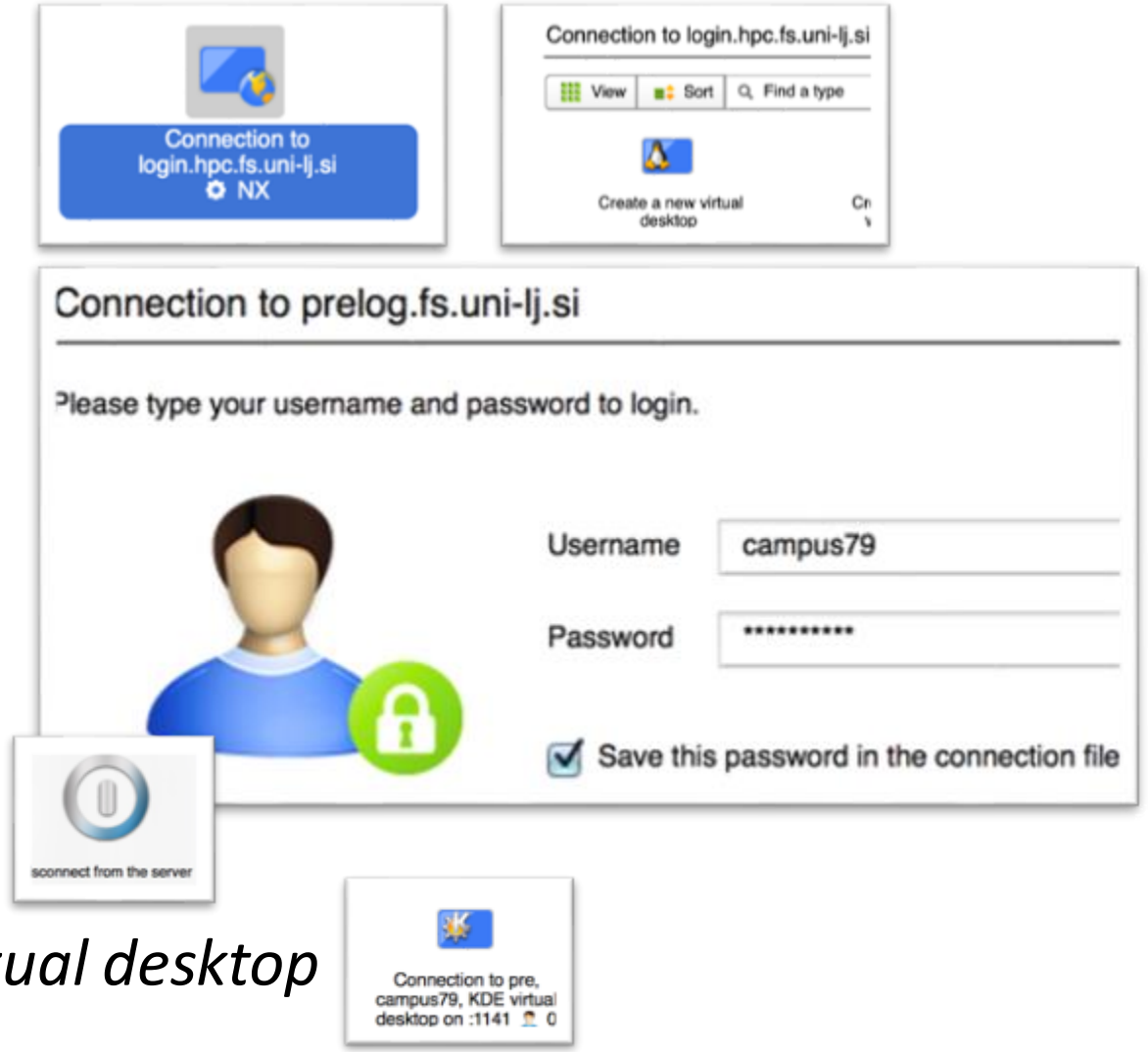
II. *Use **your** account credentials*

III. *Create **New** desktop **once***

IV. *Use the Trinity (KDE) desktop*

V. *To **Disconnect** press Ctrl+Alt+T*

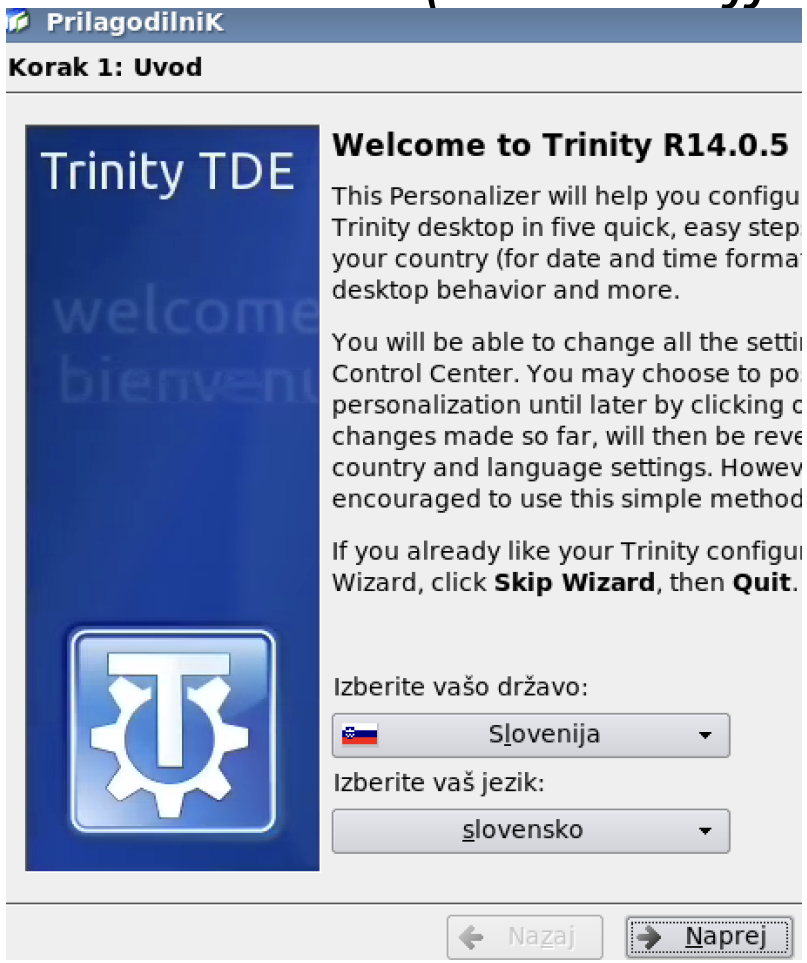
VI. *To **Reconnect** select previous virtual desktop*

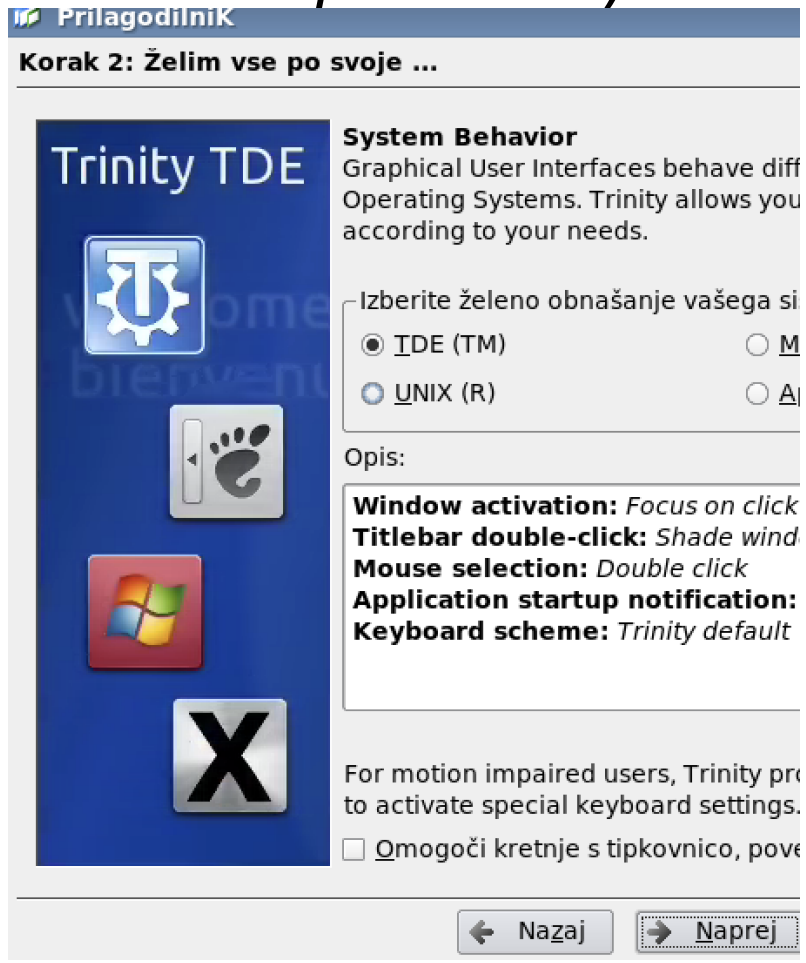


Tuning desktop with KPersonalizer for remote speed

(use less effects=slower processor)

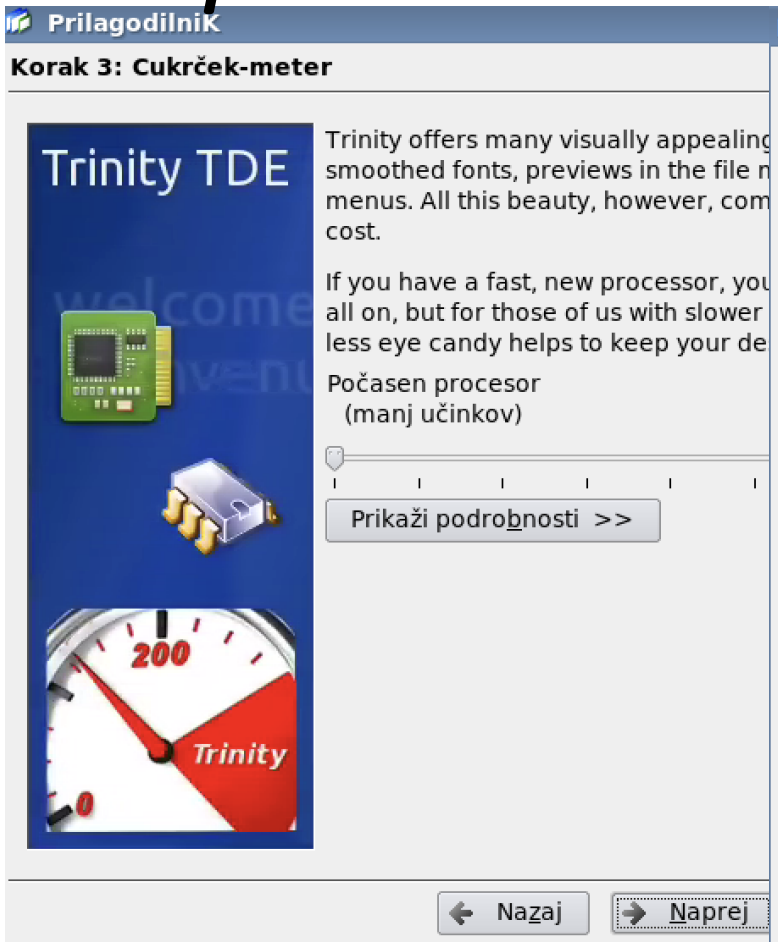
Use Trinity Control Center to setup colors for Non TDE programs: Uncheck “Enforce colors for Non-TDE programs”

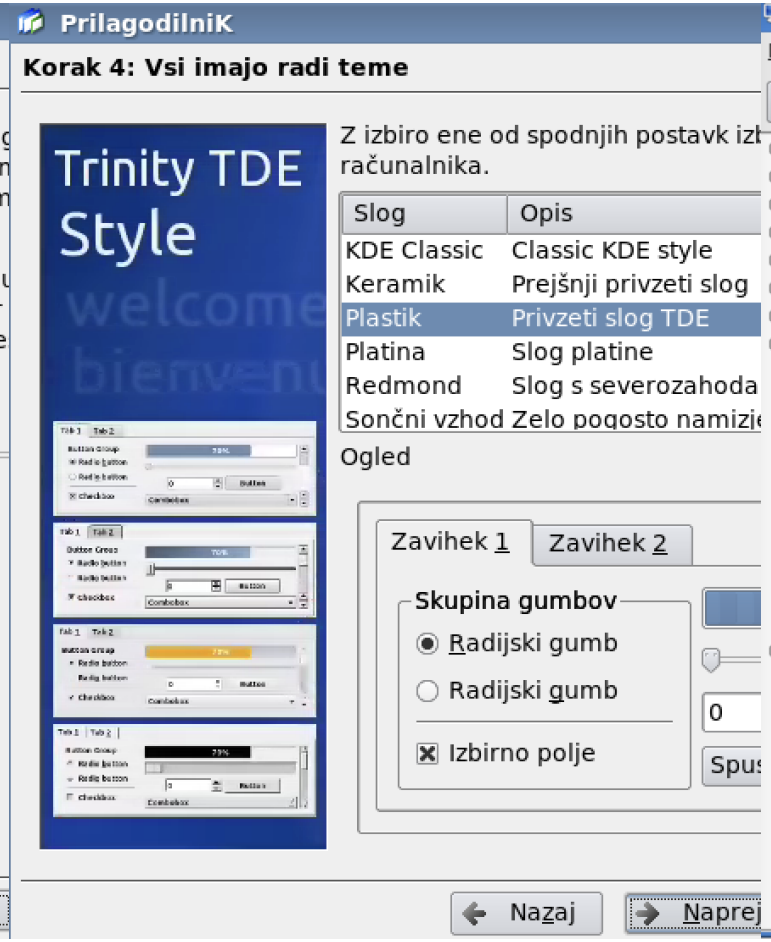


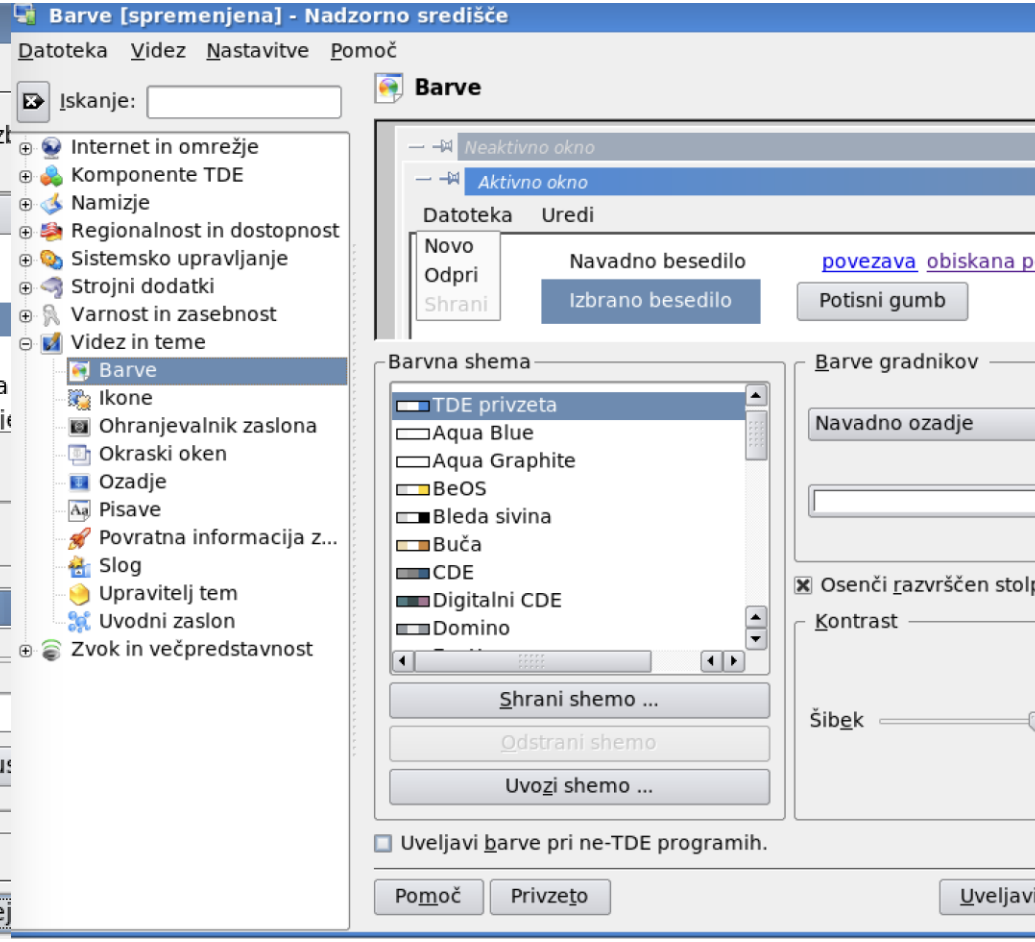


Tuning desktop with KPersonalizer for remote speed

(use less effects=slower processor)







Basic HPCFS cluster usage

- *Setting GNOME or KDE desktop locale preferences for keyboard, LANG environment*
- *Using NX client (Disconnect, Terminate, Logout)*
- *Console commands in Linux*
- *Editors for programming (emacs, gedit, kate, eclipse, vi, pico, ...) on login only!*

Modules (LUA)

- module avail
- module help/info
- module show
- module load/unload
- module list
- module purge

SLURM batch scheduler - Compiled-in OpenMPI support

- srun --nodes=N --ntasks=n cmd
- sbatch script.sh
- sinfo
- squeue
- Alias for interactive usage of nodes:

  alias node='srun -N1 --time=1:00:00 --pty bash -i'

Accessing HPCFS

Useful SLURM commands

- sbatch myjob.slurm

```
#!/bin/bash
#SBATCH -J "MyFEM 1" # Ime naloge
#SBATCH -n 4 # Stevilo procesov
#SBATCH -N 2 # Stevilo vozlisc
#SBATCH -p rome # particija
#SBATCH --mem=0 # ves spomin vozlisca
#SBATCH --time=0:10:0 # najvec 10 minut
#SBATCH -o output.log # Zaslonski izpis
#SBATCH -e error.log # Mozne napake programa

module purge
module load Eigen/3.4.0-GCCcore-12.2.0
module load VTK/9.2.6-foss-2022b
module load OpenMPI/4.1.5-GCC-12.2.0
```

- sinfo or sinfo -s
- squeue or squeue -u $USER
- scancel <jobid>
- scontrol show --details job <jobid>
- squeue --user=$USER --start --iterate=60

HPCFS – software overview

avaliable software

- ANSYS Multiphysics
- ANSYS CFX, Fluent, …
- OpenFOAM CFD, Elmer Multiphysics
- ParaView postprocessor, SALOME
- Compilers: GNU, Intel F90, CC
- R, Mathematica, MATLAB, Python
- Parallel environments: OpenMP, OpenMPI, IntelMPI
- Libraries: BLAS, BLACS, FFTW, GOTO, MUMPS, NetCDF, HDF5, Sparsekit, Scalapack, Tensorflow

HPCFS – software overview

Module environment

- Module concept is available on most supercomputers
- It simplifies the use of different software (versions) in a controlled way.
- Settings for each SW (version) are encapsulated into "environment modules" maintained by the module system that updates environment variables such as search paths ($PATH), dynamic loader paths ($LD_LIBRARY_PATH).

HPCFS – software overview

Basic module commands

- To get list currently loaded modules on the node

[johndoe@cn62 ~]$ **module list**

No modules loaded

- To list all available modules on HPC

[johndoe@cn62 ~]$ **module avail**

```
------------------------------------------------------------------------------ /opt/pkg/modules/all ------------------------------
   ANSYS/20.1                                                    Tcl/8.6.9-GCCcore-8.2.0
   ANSYS/21.R1                                        (D)        Tcl/8.6.9-GCCcore-8.3.0
   ANSYS/21.1                                                    Tcl/8.6.10-GCCcore-9.3.0
   ANSYS/2021R2                                                  Tcl/8.6.10-GCCcore-10.2.0                                  (D)
   ATK/2.22.0-foss-2016b                                         TensorFlow/1.10.1-foss-2018b-Python-3.6.6
   ATK/2.28.1-fosscuda-2018b                                     TensorFlow/1.10.1-fosscuda-2018b-Python-2.7.15
   ATK/2.32.0-GCCcore-8.2.0                           (D)        TensorFlow/1.13.1-foss-2019a-Python-3.7.2
   ATLAS/3.10.2-GCC-5.4.0-2.26-LAPACK-3.6.1                      TensorFlow/1.13.1-fosscuda-2019a-Python-3.7.2
   Arrow/0.17.1-foss-2020a-Python-3.8.2                          TensorFlow/2.0.0-foss-2019a-Python-3.7.2
   Autoconf/2.69-foss-2016b                                      TensorFlow/2.0.0-fosscuda-2019b-Python-3.7.4                (D)
   Autoconf/2.69-GCC-4.9.3-2.25                                  Tk/8.6.5-foss-2016b
```

HPCFS – software overview commands - keyword

- To search for a module starting with some **keyword**

[johndoe@cn62 ~]$ **module keyword Ansy**

The following modules match your search criteria: "Ansy"

ANSYS: ANSYS/2021R2, ANSYS/2022R1, ANSYS/2022R2, ANSYS/2023R1, ANSYS/2024R1, ANSYS/2024R2

Ansys offers a comprehensive software suite that spans the entire range of physics, providing access to virtually any field of engineering simulation that a design process requires.

HPCFS – software overview

commands - whatis

- To see details about some module

[johndoe@cn62 ~]$ **module whatis** MATLAB

MATLAB/2023a : Description: MATLAB is a high-level language and interactive environment that enables you to perform computationally intensive tasks faster than with traditional programming languages such as C, C++, and Fortran.

MATLAB/2023a : Homepage: https://www.mathworks.com/products/matlab

MATLAB/2023a : URL: https://www.mathworks.com/products/matlab

HPCFS – software overview commands - spider

- For slightly more details about given module

  [johndoe@cn62 ~]$ **module spider MATLAB**

MATLAB:

Description:

MATLAB is a high-level language and interactive environment that enables you to perform computationally intensive tasks faster than with traditional programming languages such as C, C++, and Fortran.

Versions:

MATLAB/2019a

MATLAB/2022b

MATLAB/2023a

HPCFS – software overview commands - show

- For even more details about given module

  [johndoe@cn62 ~]$ **module show MATLAB**

```
   /opt/pkg/modules/all/MATLAB/2023a.lua:
-------------------------------------------------------------------------------
help([[
Description
===========
MATLAB is a high-level language and interactive environment
 that enables you to perform computationally intensive tasks faster than with
 traditional programming languages such as C, C++, and Fortran.


More information
================
 - Homepage: https://www.mathworks.com/products/matlab
]])
whatis("Description: MATLAB is a high-level language and interactive environment
 that enables you to perform computationally intensive tasks faster than with
 traditional programming languages such as C, C++, and Fortran.")
whatis("Homepage: https://www.mathworks.com/products/matlab")
whatis("URL: https://www.mathworks.com/products/matlab")
conflict("MATLAB")
prepend_path("CMAKE_PREFIX_PATH","/opt/pkg/software/MATLAB/2023a")
prepend_path("PATH","/opt/pkg/software/MATLAB/2023a/bin")
setenv("EBROOTMATLAB","/opt/pkg/software/MATLAB/2023a")
setenv("EBVERSIONMATLAB","2023a")
setenv("EBDEVELMATLAB","/opt/pkg/software/MATLAB/2023a/easybuild/MATLAB-2023a-easybuild-devel")
prepend_path("PATH","/opt/pkg/software/MATLAB/2023a")
prepend_path("LD_LIBRARY_PATH","/opt/pkg/software/MATLAB/2023a/runtime/glnxa64")
prepend_path("LD_LIBRARY_PATH","/opt/pkg/software/MATLAB/2023a/bin/glnxa64")
prepend_path("LD_LIBRARY_PATH","/opt/pkg/software/MATLAB/2023a/sys/os/glnxa64")
setenv("_JAVA_OPTIONS","-Xmx2048m")
```

HPCFS – software overview

Most important commands

[johndoe@cn62 ~]$ **module available MATLAB**

---------- /opt/pkg/modules/all ----------

MATLAB/2019a MATLAB/2022b MATLAB/2023a (D)

[johndoe@cn62 ~]$ **module load MATLAB/2023a**

[johndoe@cn62 ~]$ **module list**

Currently Loaded Modules:

1) MATLAB/2023a

[johndoe@cn62 ~]$ **module unload MATLAB/2023a**

If you want to remove all the currently loaded modules and clean the environment variables:

[johndoe@cn62 ~]$ **module purge**

Some other modules

- Intel compiler

  [johndoe@cn62 ~]$ module load intel

  [johndoe@cn62 ~]$ module load iimpi # with Intel MPI

- GCC compiler and MPI Free and Open Source Software environment

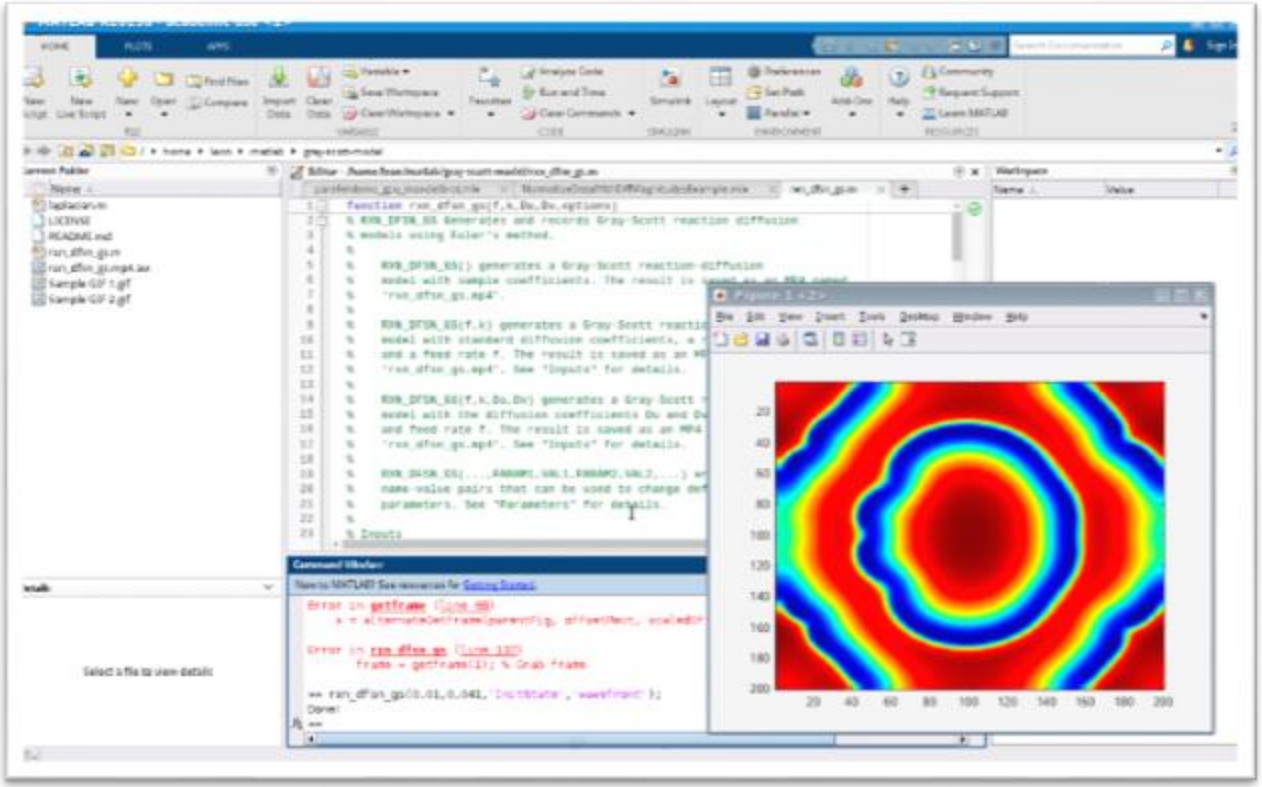
  [johndoe@cn62 ~]$ module available foss/

  [johndoe@cn62 ~]$ module load foss/2023b

Avaliable software: Matlab

module load MATLAB

- env --unset LD_PRELOAD TMOUT=600 srun --time=2:00:00 --partition=haswell ${DISPLAY:+--x11} --pty matlab



Avaliable software: Mathematica

ml Mathematica

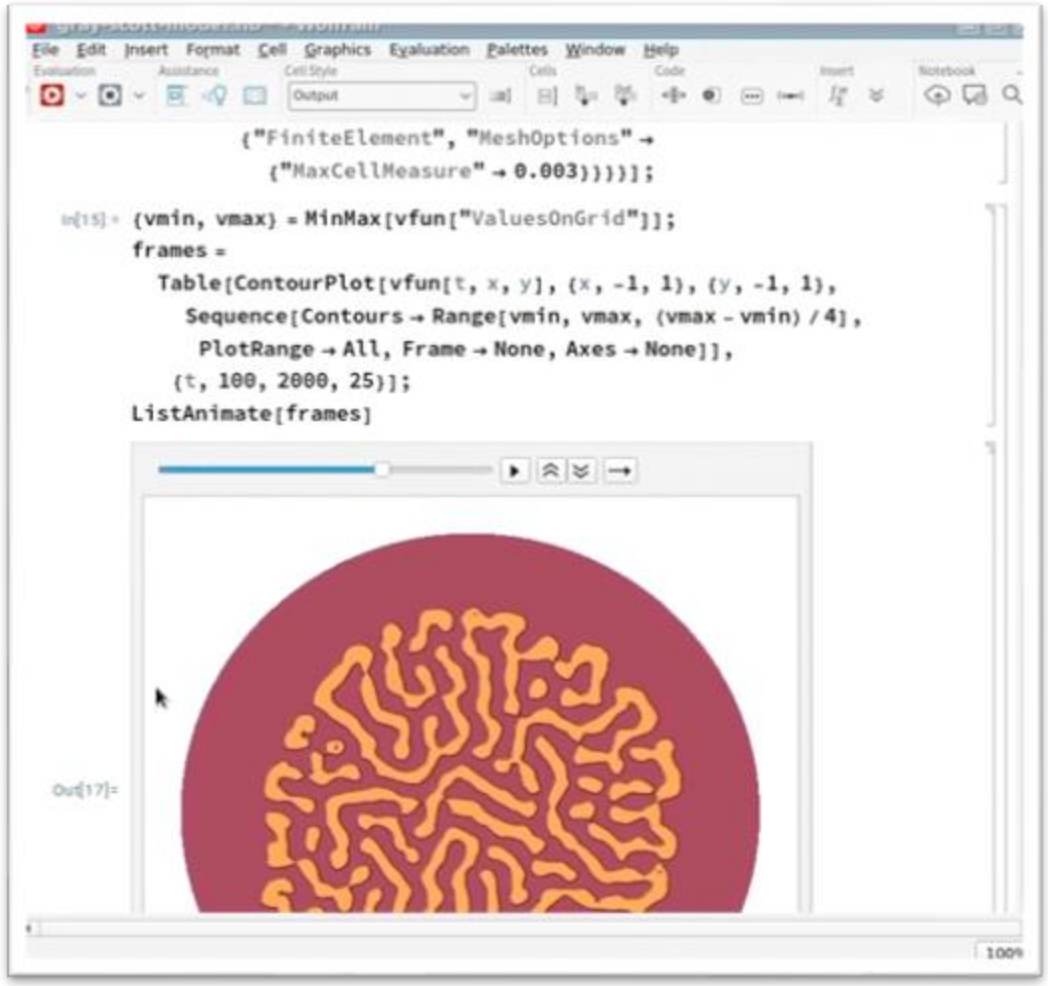
- Start with
  mathematica
  wolframnb # 14.2
- For the first time an activation window pops out suggesting activation through web. Please find a button **Other ways to activate** in the bottom of the dialog and thenselect **Activate through a Wolfram network license server**and for server enter **flex.hpc** and press **Activate**.

Parallel execution on a compute node(s) [version 14.2]

Use *Edit->Preferences->Kernels->Kernel Configuration Editor* and modify example.com to Class *SshKernels* and rename it to the compute node allocated within terminal with

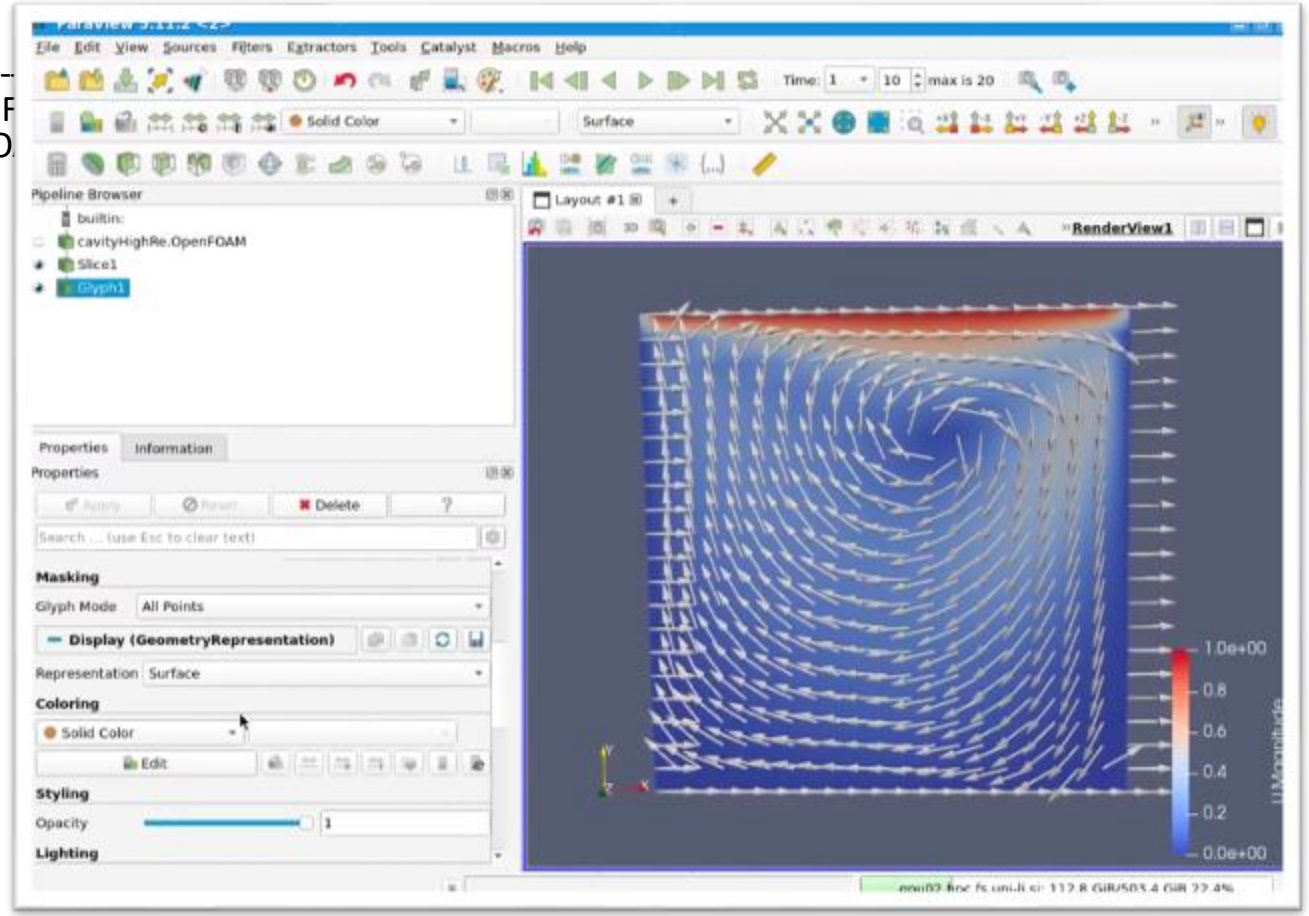
salloc -N 1

For *Default Kernels* select that configuration as a *Defult kernel for RemoteEvaluate*



Avaliable software: OpenFOAM

- $ ml av OpenFOAM
  ---------- /opt/pkg/modules/all ----------
  OpenFOAM/v2206-foss-2022a OpenFOAM/v2412-foss-2023a OpenF
  OpenFOAM/v2406-foss-2023a OpenFOAM/10-foss-2022a OpenFO
  (D)
- ml OpenFOAM
- ml show OpenFOAM
- source $FOAM_BASH
- env|grep FOAM|cut -c -40|head
- ls $FOAM_TUTORIALS/incompressibleFluid
- cp -r $FOAM_TUTORIALS/incompressibleFluid/cavity $HOME
- cd cavity
- ./Allrun
- cd cavityHighRe
- env LD_PRELOAD=/lib64/libcrypto.so:$LD_PRELOAD paraFoam



Avaliable software: Ansys

- Documentation available online:
  - https://ansyshelp.ansys.com/public
  - access loaded version from command line

    $ module load ANSYS/2024R2

    $ anshelp

- Two ways to run: **interactive** or **batch mode**
  - intractive with GUI interface on (login) node
  - batch can be scheduled – preferred use on clusters

Avaliable software: Ansys Workbench

- command:
  - runwb2
  - interactive job on compute node*:

    $ haswell runwb2

    $ rome runwb2

- batch script:
  - runwb2 -b -F file.wbpj -?...

*very version dependent, some don't work

Avaliable software: Ansys CFX

- commands:
  - cfx5
  - cfx5pre
  - cfx5solve
  - cfx5post
- batch script:

```
#!/bin/bash
#SBATCH --export=ALL,LD_PRELOAD=
#SBATCH --ntasks-per-core=1
#SBATCH --ntasks=12  # total number of cores requested
#SBATCH --nodes=2  # number of nodes
#SBATCH --job-name=cfx-test
#SBATCH --error=sbatch.err
#SBATCH --output=sbatch.out
#SBATCH --time=0-00:05  # time limit days-hh:mm
#SBATCH --mem=0G
module purge
module load ANSYS/2024R2

MACHINES=$(srun -l hostname -s | sort -n | uniq -f 1 -c | awk '{print $3 "*" $1}' | paste -sd ',')
DEF=StaticMixer.def

###  single node run
#cfx5solve -def $DEF -part $SLURM_NTASKS -start-method "Open MPI Local Parallel"

###  distributed run
cfx5solve -def $DEF -par-dist $MACHINES -start-method "Open MPI Distributed Parallel"
```

Avaliable software: Ansys Fluent

- commands:
  - fluent
- batch script:

```
#!/bin/bash
#SBATCH --export=ALL,LD_PRELOAD=
....
#SBATCH --time=0-00:05
module purge
module load ANSYS/2024R2

###  checkpointing fluent:
###  'touch check-fluent' ...save result and continue
###  'touch exit-fluent' ...save result and exit

JOURNAL=test.jou

###  distributed run
MACHINES=$(srun -l hostname -s | sort -n | uniq -f 1 -c | awk '{print $3 ":" $1}' |
paste -sd ',')

fluent 3ddp -g -t$SLURM_NTASKS -cnf=$MACHINES -mpi=intelmpi -i $JOURNAL

###  single node run
#fluent 3ddp -g -t$SLURM_NTASKS -i $JOURNAL
```

Avaliable software: Ansys MAPDL

- commands:
  - mapdl
  - ansys
  - ansys242
- batch script:

```
#!/bin/bash
#SBATCH --export=ALL,LD_PRELOAD=
#SBATCH --ntasks-per-core=1
#SBATCH --ntasks=12  # total number of cores requested
#SBATCH --nodes=2  # number of nodes
#SBATCH --job-name=mapdl-test
#SBATCH --error=sbatch.err
#SBATCH --output=sbatch.out
#SBATCH --time=0-00:05  # time limit days-hh:mm
#SBATCH --mem=0G

module purge
module load ANSYS/2024R2

mapdl -dis -b -m 10000 -np $SLURM_NTASKS -mpi intelmpi2018 -i bench07.mac -o bench07.out
```

Avaliable software: Ansys LS-Dyna

- commands:
  - mapdl
  - ansys
  - ansys242
- batch script:

```
#!/bin/bash
#SBATCH --export=ALL,LD_PRELOAD=
…
#SBATCH --signal=B:USR1  # default @60s before time limit

module load ANSYS/2024R2
export LSTC_LICENSE=ansys

FILENAME=000_yaris_dynamic_roof_crush_01

ansysbin=$(dirname $(which lsdyna))
lsdyna_e=$ansysbin/linx64/lsdyna_dp_mpp.e
…..

srun --mpi=pmi2 $lsdyna_e memory=250M memory2=100M \
                          i=$FILENAME.k \
                          o=$FILENAME.out \
                          g=$FILENAME.d3plot \
                          f=$FILENAME.d3thdt &
```

Using SLURM (interactivelly) and Message Passing Interface (MPI)

```
[leon@viz mpi]$ module purge && module load foss/2019a
[leon@viz mpi]$ cat hello.f90
program hello
        use mpi
        integer rank, size, ierror, strlen, status(MPI_STATUS_SIZE)
        character(len=MPI_MAX_PROCESSOR_NAME) :: hostname

        call MPI_INIT(ierror)
        call MPI_COMM_SIZE(MPI_COMM_WORLD, size, ierror)
        call MPI_COMM_RANK(MPI_COMM_WORLD, rank, ierror)
        call MPI_GET_PROCESSOR_NAME( hostname, strlen, ierror )
        print*, trim(hostname), rank, size
        call MPI_FINALIZE(ierror)
end
[leon@viz mpi]$ mpif90 hello.f90
[leon@viz mpi]$ LD_PRELOAD= srun -n 4 --tasks-per-node=2 --kill-on-badexit
--partition=haswell ./a.out
cn80    2       4
cn79    0       4
cn80    3       4
cn79    1       4
```

OpenMP

```
#include <stdio.h>
#include <math.h>
#define N 1000000
int main()
{
        double area = 0.0;
        #pragma omp parallel for reduction(+:area)
        for(int i = 0; i < N; i++)
                {
                        double x = (i+0.5)/N;
                        area += sqrt(1.0 - x*x);
                }
        printf("Pi : %14lf\n", 4.0*area/N);
        return 0;
}
[leon@cn36 pi]$ module purge && module load foss/2019a
[leon@cn36 pi]$ gcc -fopenmp pi-openmp.c -lm -o pi-openmp
[leon@cn36 pi]$ OMP_NUM_THREADS=4 ./pi-openmp
```

Thank you for attention!

EuroHPC Joint Undertaking

REPUBLIC OF SLOVENIA
MINISTRY OF HIGHER EDUCATION, SCIENCE AND INNOVATION

This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 101101903. The JU receives support from the Digital Europe Programme and Germany, Bulgaria, Austria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Greece, Hungary, Ireland, Italy, Lithuania, Latvia, Poland, Portugal, Romania, Slovenia, Spain, Sweden, France, Netherlands, Belgium, Luxembourg, Slovakia, Norway, Türkiye, Republic of North Macedonia, Iceland, Montenegro, Serbia.