HPC User Environment

Dirk Schmidl

schmidl@rz.rwth-aachen.de

Center for Computing and Communication
RWTH Aachen University
22.03.2010

HPC User Environment

- Login to Linux
- Login to Windows
- Cluster Environment on Linux
  - Compilers
  - MPI implementations
  - Independent Software Vendor (ISV) products
  - the module system
- Cluster Environment on Windows
- Storage
- using the Batch system on Linux (Oracle Grid Engine)
- using the Batch system on Windows (Microsoft HPC Job Manager)

cluster@rz.rwth

Linux:

~ 250 Harpertown nodes

- **2x Xeon E5450 @ 3.00 GHz**
- **16 GB RAM**
- **DDR Infiniband**

~ 200 Nehalem nodes

- **2x Xeon E5570 @ 2.93 GHz**
- **24 GB RAM**
- **QDR Infiniband**

~ 50 Barcelona Opteron nodes

- **4x Opteron Processor 8356 @ 2.30 GHz**
- **32 GB RAM**
- **DDR Infiniband**

cluster@rz.rwth

Windows:
~ 25 Harpertown nodes
- **2x Xeon E5450 @ 3.00 GHz**
- **16 GB RAM**
- **DDR Infiniband**

~ 5 Nehalem nodes
- **2x Xeon E5570 @ 2.93 GHz**
- **48 GB RAM**
- **QDR Infiniband**

Nodes can be switched between Windows and Linux if needed.

Currently we had a job using 64 dedicated Windows nodes.

Usage Model

Login to Frontend nodes for interactive use:

e.g.: program development, debugging, using GUIs, small test jobs

LINUX	WINDOWS
cluster, cluster-linux-opteron, cluster-linux, cluster-linux-nehalem cluster-linux-dunnington cluster-linux-xeon	cluster-win (automatically scheduled to 4 different machines for load balancing)

Use Backend nodes via our batch systems for large calculations.

PRO:

- nodes are not overloaded with too many jobs
- systems are also used at night und on the weekend

CON:

- jobs sometimes need to wait before they can start

Login to Linux: console

SSH (command line) frontends

SSH Comes by default with Unix and Linux

```
ssh [-Y] user@cluster.rz.rwth-aachen.de
scp[[user@]host1:]file1 [...] [[user@]host2:]file2
```

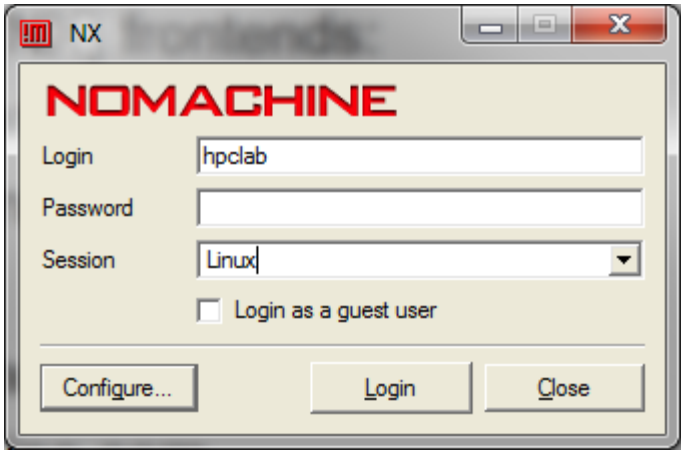
Get Windows clients from

Putty: http://www.chiark.greenend.org.uk/~sgtatham/putty/

ssh for Windows: ftp://ftp.cert.dfn.de/pub/tools/net/ssh

Login to Linux: NX

- Tool to start graphical remote sessions
- client available for Windows/Linux/Solaris/MacOSX
  www.nomachine.com
- Server available for Linux / Windows
- only Linux Server available here

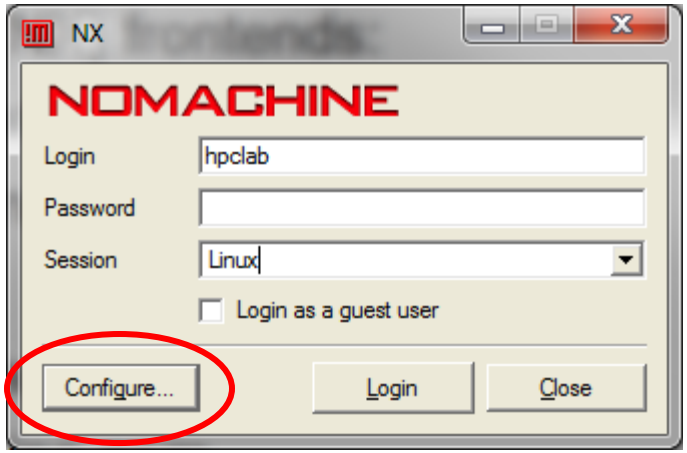


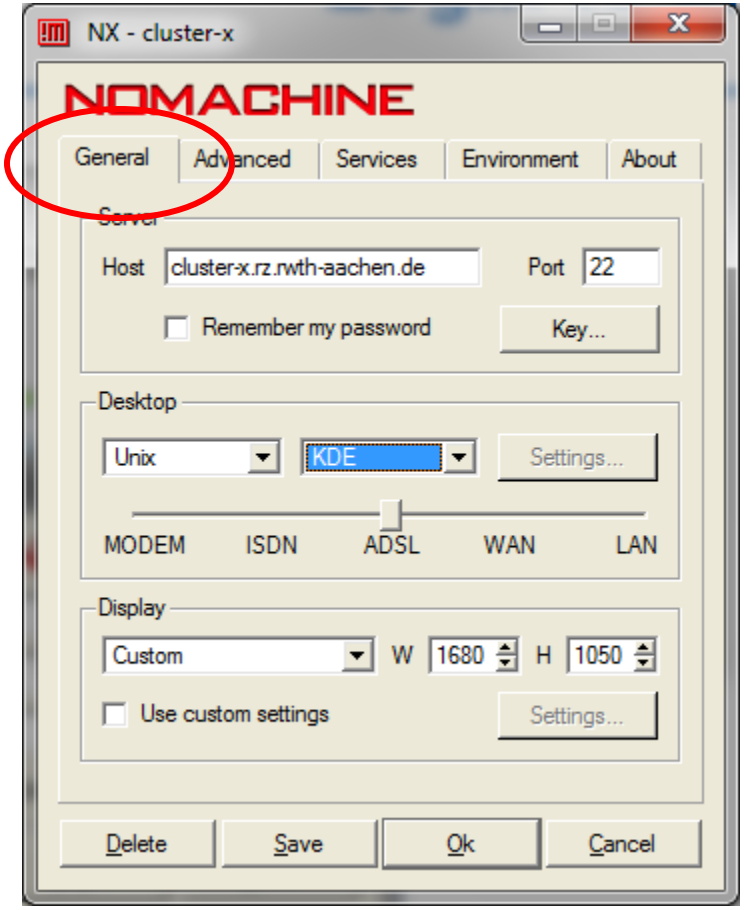
Login to Linux: NX

Graphical login (NX) frontends:

cluster-x.rz.rwth-aachen.de

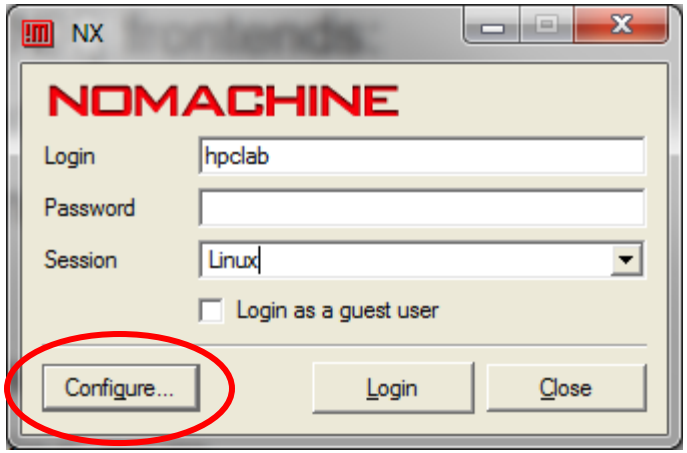
cluster-x2.rz.rwth-aachen.de

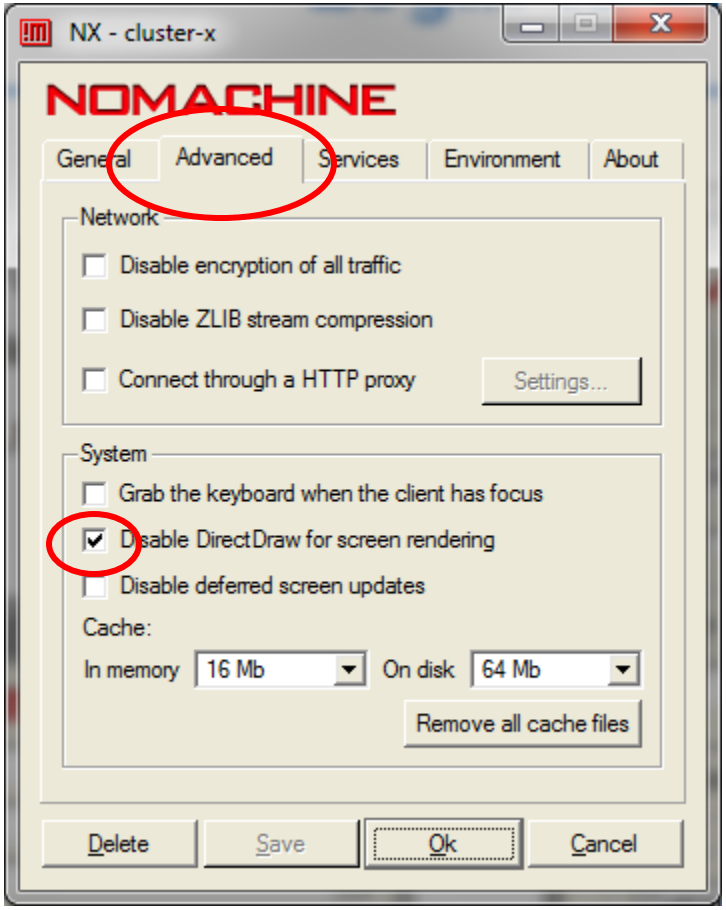




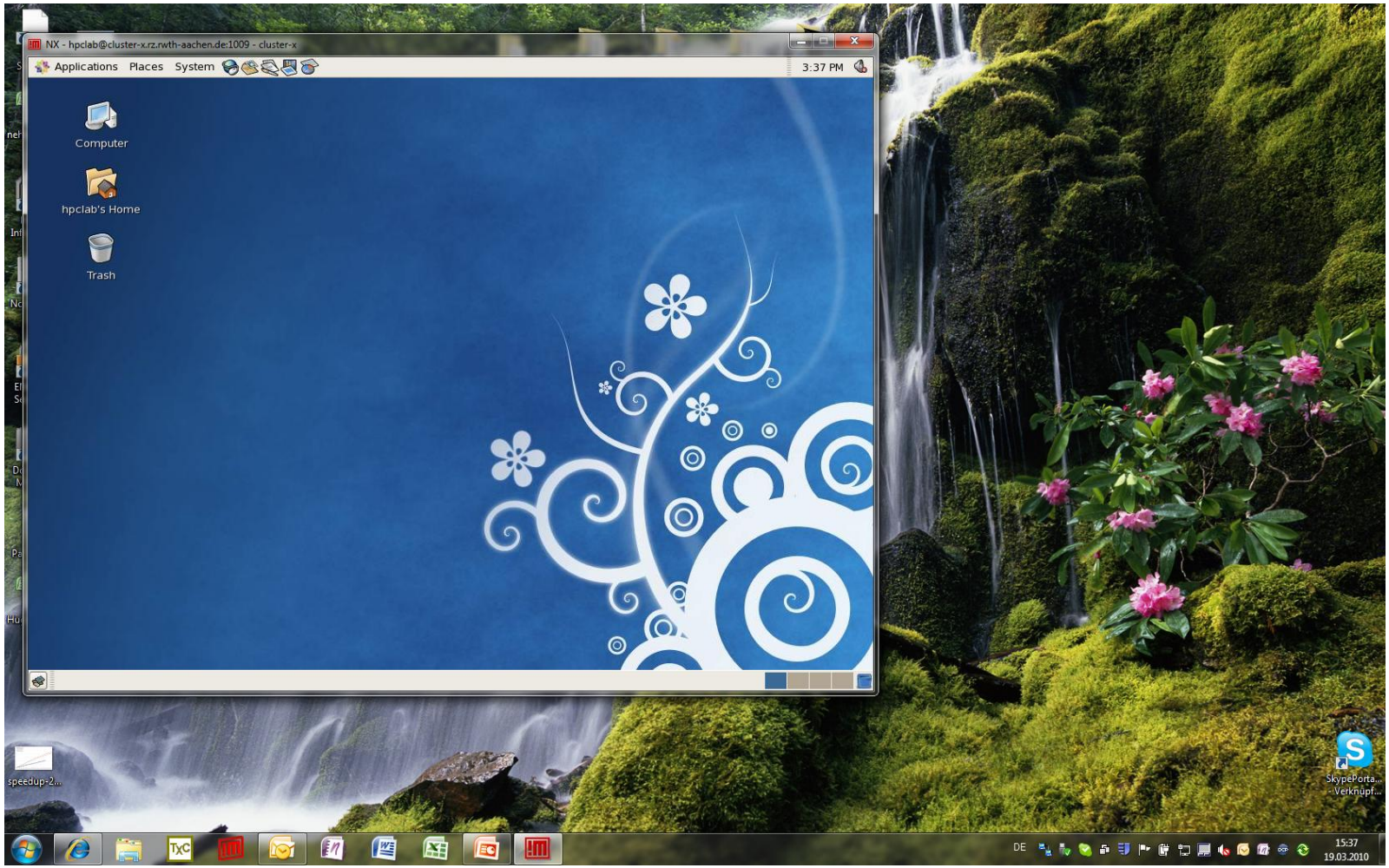
Login to Linux: NX

Tip: Setting this option improves performance.





Login to Linux: NX



DEMO

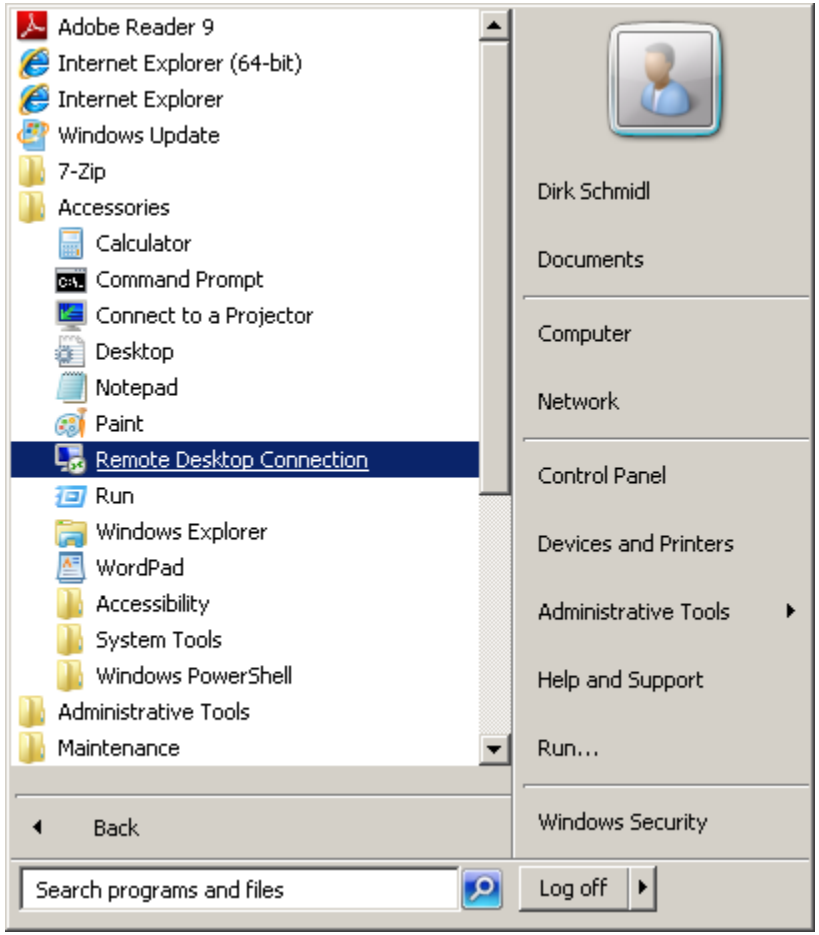
Login to Windows: Remote Desktop Connection

Frontend nodes:

cluster-win.rz.rwth-aachen.de

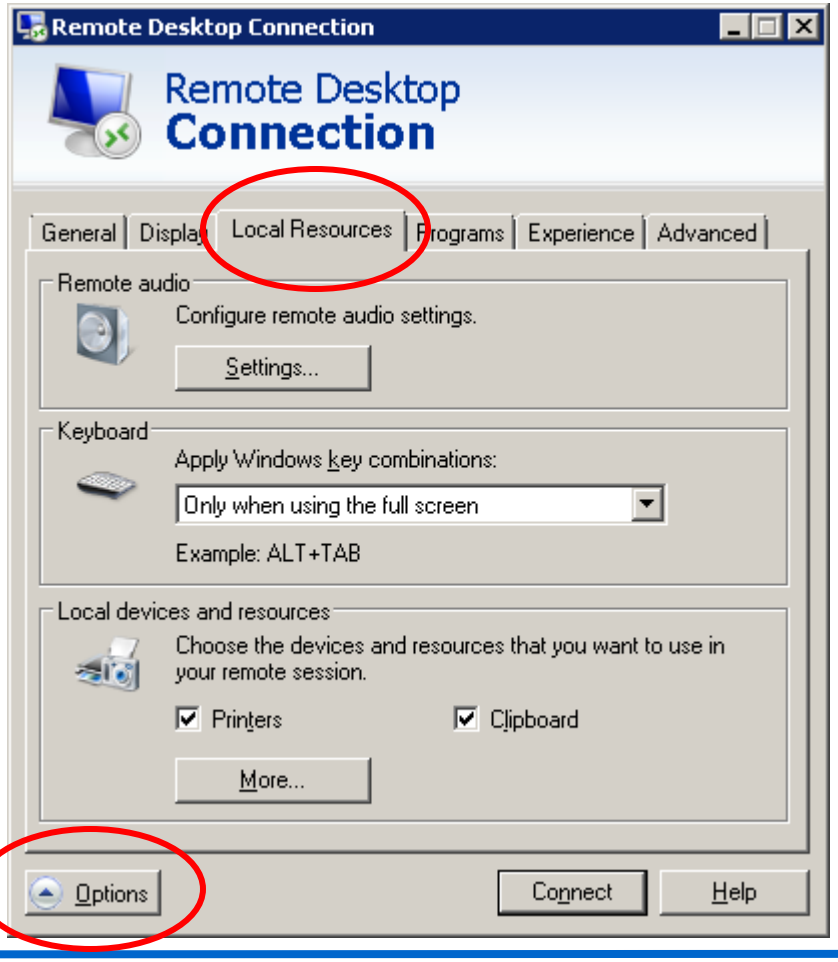
load balancing:

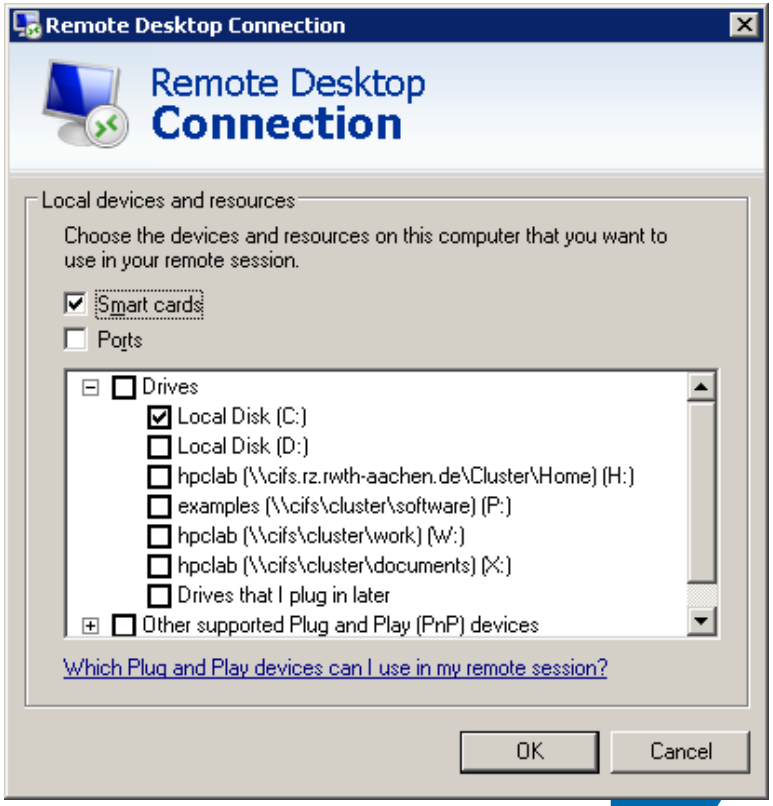
forward to one of 4 machines



Login to Windows: Remote Desktop Connection

sharing local resources





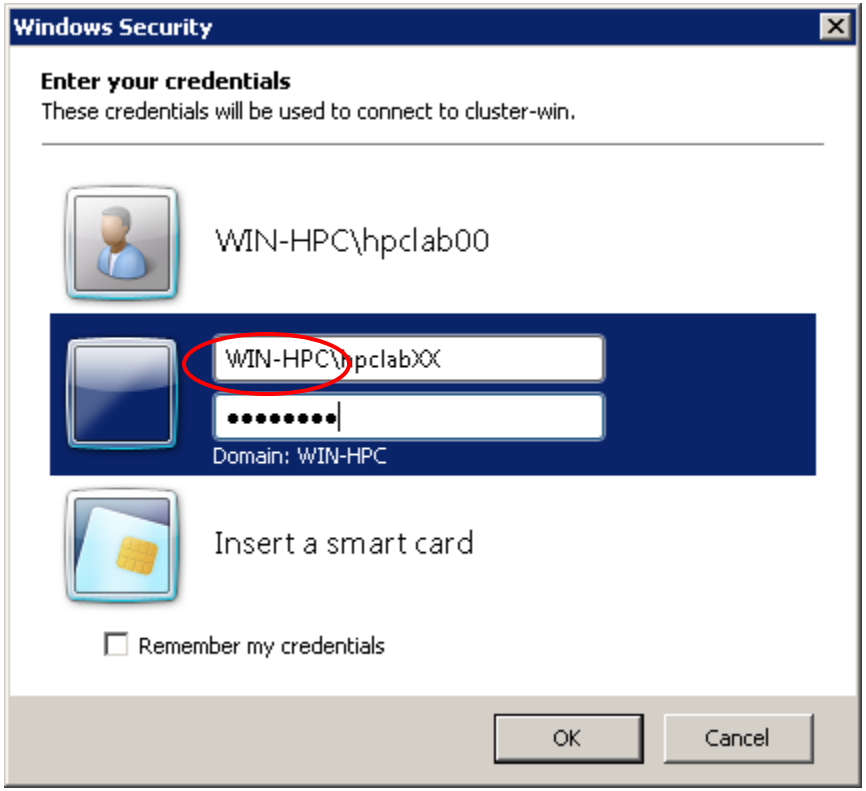
Login to Windows: Remote Desktop Connection

Frontend nodes:

cluster-win.rz.rwth-aachen.de

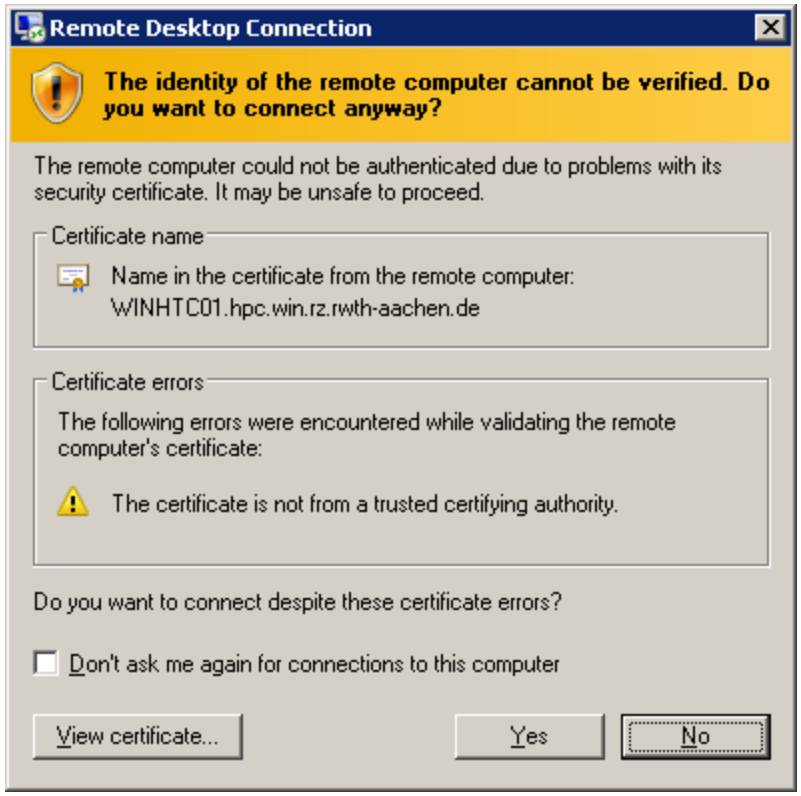
load balancing:

forward to one of 4 machines



Login to Windows: Remote Desktop Connection

NOTE: Security warning might occur multiple times due to load balancing.



DEMO

Login to Windows: rdesktop

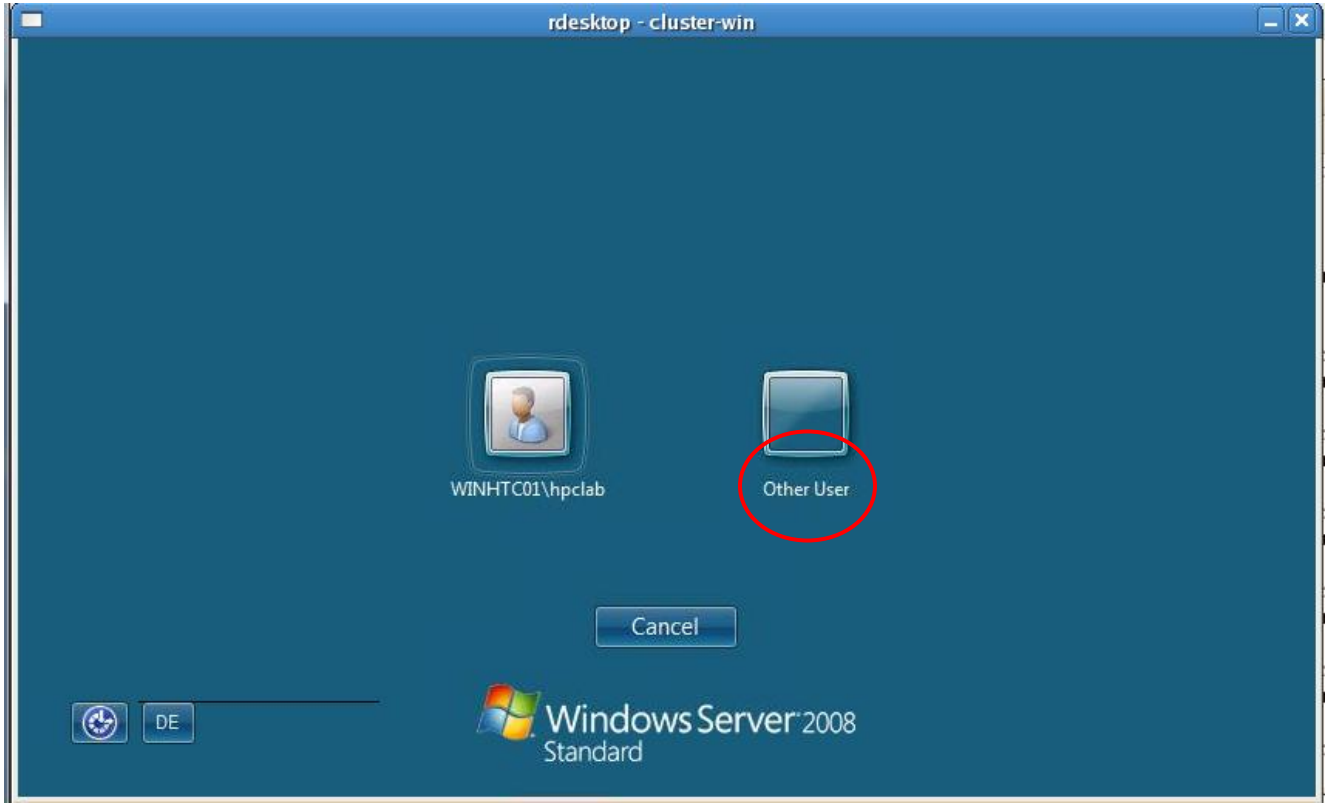
From Linux:

r*desktop [options] cluster-win.rz.rwth-aachen.de*

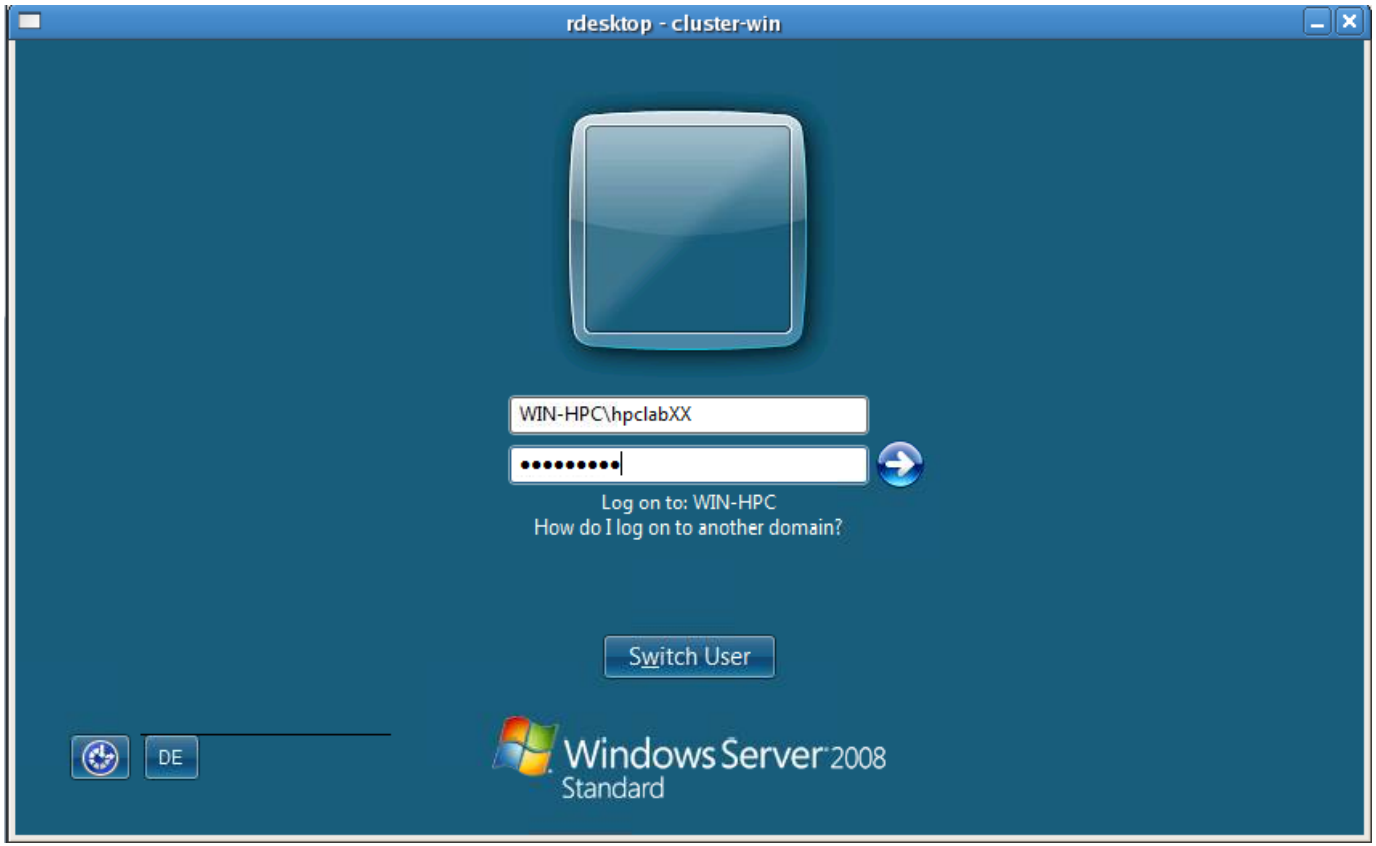
option	short description	example
-u	username	-u hpclabXY
-k	keyboard map	-k en
-f	fullscreen	-f
-g	size (geometry)	-g 90% | -g 1024x768
-d	choose domain	-d WIN-HPC
-r disc:	redirect disc	-r disk:home=/home/hpclab

Login to Windows: rdesktop

Take care to choose WIN-HPC domain.



Login to Windows: rdesktop



Maybe you must login *twice* due to the load balancing mechanism.

DEMO

Linux Environment

CentOS 5.3 (based on RedHat Linux)

Default shell: the Z shell (zsh)

also available: bash, ksh, …

need a trick to use modules in bash (see primer)

Show environment limits:

```
ulimit -a
```

Max. time a process can run

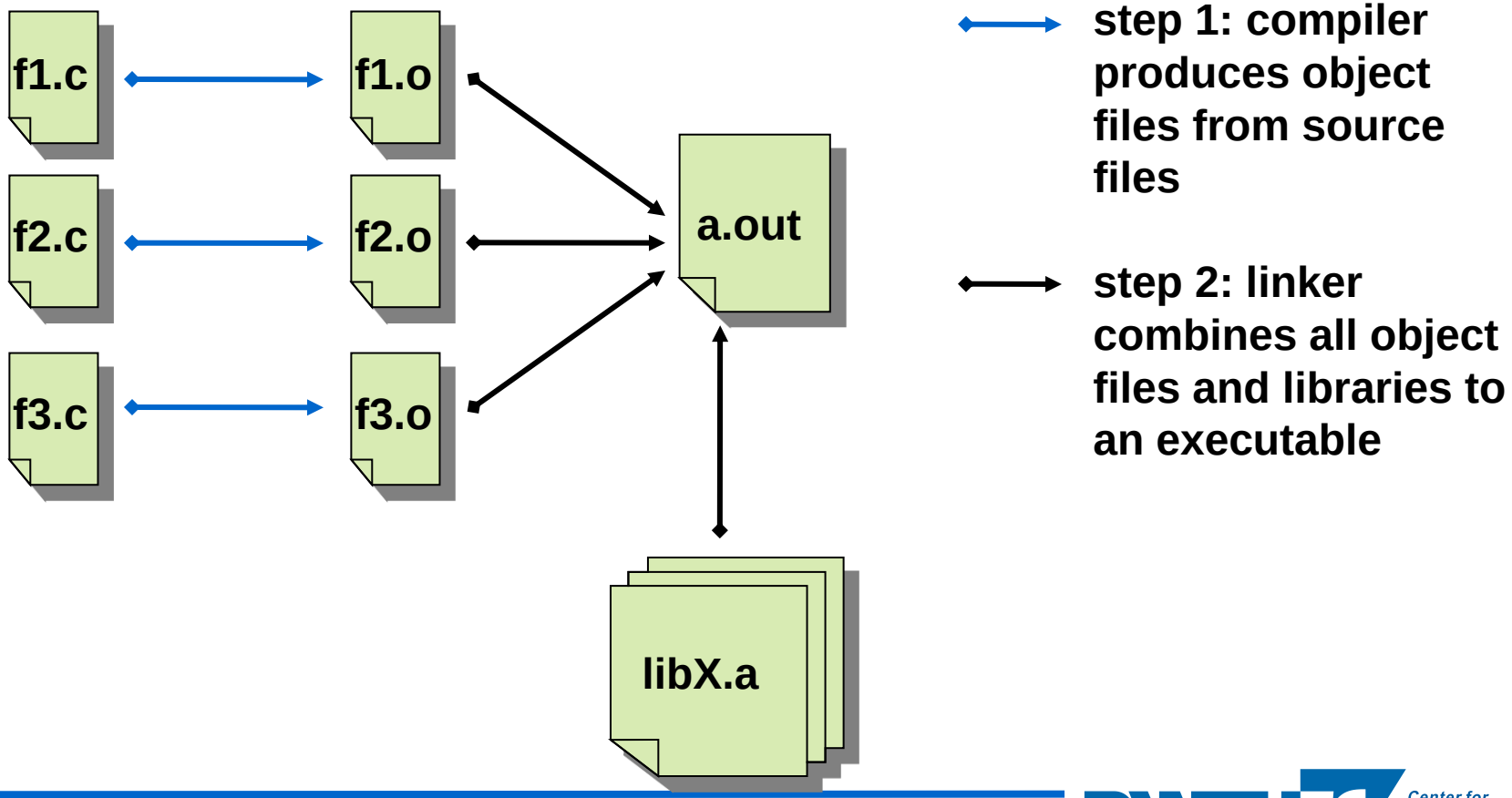
```
ulimit -t seconds
```

Max. stacksize

```
ulimit -s kilobytes
```

Compilers

Compilers are used to build executables out of source files.



Compilers

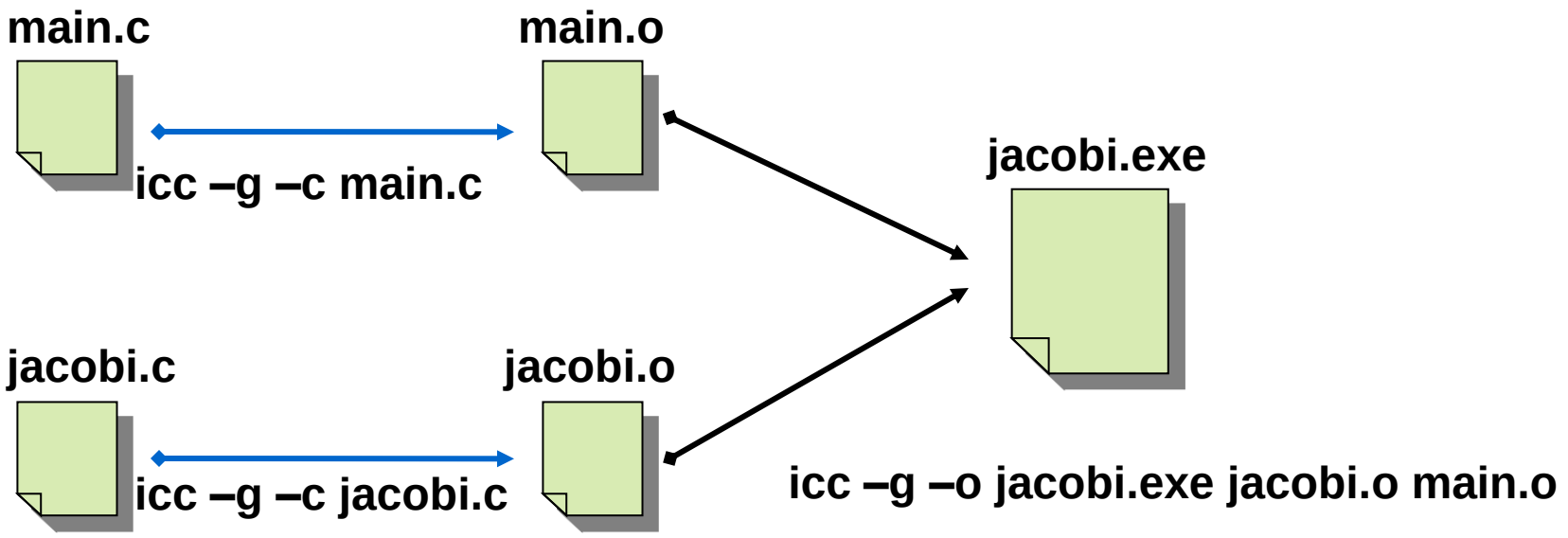
Intel Compiler usage: (icc | icpc | ifort)

icc [options] file1 [file2 …]

option	description
-o <file>	specifiy output file name
-c	only compilation, no linking
-g	generate debug information in output file
-O[n]	set optimization level n
-openmp	generate multithreaded code based on OpenMP pragmas
-x[SSE4.2|SSSE3 |SSE3|SSE2]	optimize using SSE instructions note: the processor must support the SSE level

Compilers

example:



Compilers

example run with different optimization flags for jacobi code

options	MFLOP/s
-g	549
-g –O1	1334
-g –O2	1982
-g –O3	2047
-g –O3 –xSSE4.2	2183

using the right compiler options is essential for performance

Compilers

Available Compilers:

intel compilers

- v11.0
- v11.1

gcc

- v3.4
- v4.2
- v4.3
- v4.4

Oracle Solaris Studio
for Linux
(formerly Sun Studio)

- v12.0
- v12.1

pgi

- v10.0
- v10.1
- v10.2
- v10.3

a lot of different compilers and versions available

Compilers

Many Compilers are nice to have, but its hard to recognize all options.

e.g.: enable OpenMP

- gcc: -fopenmp
- intel: -openmp
- oracle: -xopenmp
- pgi: -mp[=options]

Compilers

On our systems some environment variables are set to make this easier.

Compilers:

$CC	C Compiler
$CXX	C++ Compiler
$FC	Fortran Compiler

Optimization environment variables

$FLAGS_FAST	high optimization
$FLAGS_FAST_NO_FPOPT	high optimization without floating point changes

OpenMP + Debug environment variables

$FLAGS_OPENMP	activate OpenMP
$FLAGS_DEBUG	enable debug mode

Compilers

Example:

previous line for intel compiler	for all compilers
icc –g –c main.c	$CC $FLAGS_DEBUG –c main.c
icc –g –c jacobi.c	$CC $FLAGS_DEBUG –c jacobi.c
icc –g –o jacobi.exe jacobi.o main.o	$CC $FLAGS_DEBUG –o jacobi.exe jacobi.o main.o

MPI

MPI = Message Passing Interface

Available MPIs:

- Oracle Message Passing Toolkit (based on OpenMPI)
  - **v8.1**
  - **v8.2**
  - **v8.2.1**
- Intel MPI (based on MPICH2)
  - **v3.1**
  - **v3.2**

Mainly a library and some tools to compile and start parallel MPI programs.

MPI

Compiler wrapper tools for sunmpi:

mpicc / mpif90 / mpicxx

Tool to start programs on multiple machines:

mpiexec [options] <program> [args]

option	description
-np <n>	number of processes to start
-x <ENV>	pass environment variable ENV to all processes
-H <hosts>	specify hosts to start processes on
-h	print help message
-machinefile <file>	machines specifies in file are used for the processes

MPI

Variables:

$MPICC	mpi C wrapper
$MPICXX	mpi C++ wrapper
$MPIFC	mpi Fortran wrapper
$MPIEXEC	wrapper to start processes

We use a modified wrapper to achieve load balancing on some backend machines.

To see a list where the processes run, use --show option for $MPIEXEC.

ISV Software Packages

FEM
- Abaqus
- Ansys
- Hyperworks
- Ls-Dyna
- Marc/Mentat
- Nastran/Patran

Chemie
- Gaussian
- Turbomole
- Vasp
- abinit
- meep

CFD
- CFX/TASCFlow
- Fluent
- ICEM-CFD
- StarCD
- Gerris

Mathematik
- Maple
- Mathematica

Misc
- Matlab/Simulink
- Tecplot
- Siesta

Linux Environment: modules

Many compilers, many MPIs, a lot of ISV products.

The *module package* helps to manage all the software

`module help`	: print help message
`module list`	: see which modules are loaded
`module avail`	: see which modules are available

Linux Environment: modules

Load/unload a software:

```
module load <modulename>
module unload <modulename>
```

Modules can depend on other modules

e. g. special MPI module for every compiler

exchange a module:

```
module switch <moduleold> <modulenew>
```

reload all modules: (may fix the environment, often needed for NX session)

```
module reload
```

Linux Environment: modules

```
$ module avail
------------ /usr/local_rwth/modules/modulefiles/linux/x86-64/DEVELOP ------------
acumem/1.0.6                    java/1.5(default)
acumem/2009.2(default)          java/1.6
gcc/3.4                         openmpi/1.3.3(default)
gcc/4.2                         parawise/3.1(default)
gcc/4.3                         pgi/10.0
gcc/4.4                         pgi/10.1
gcc/system-default(default)     pgi/10.2
intel/11.0                      pgi/10.3(default)
intel/11.1(default)             pgi/7.1
intel/11.1.056                  pgi/8.0
intel/11.1.059                  python/2.4.2
...                             …

------------ /usr/local_rwth/modules/modulefiles/GLOBAL ------------
BETA       DEPRECATED GRAPHICS   MATH      TECHNICS   VIHPS
CHEMISTRY  DEVELOP    LIBRARIES  MISC      UNITE
```

modules

categories

Linux Environment: modules

Modules bundled in categories (MISC, CHEMISTRY…)

Appropriate category must be loaded first.

DEVELOP is loaded by default.

example: to load matlab

```
module load MISC
module load matlab
```

Find out in which category a module is:

```
module apropos modulename
```

DEMO

Data Storage

- Two different file systems
- Accessible from all nodes from Linux and Windows

$HOME or H:	$WORK or W:
less space	more space
snapshots	no snapshots
backups	no backup
for long-time data (e.g. source files, …)	for big reproducible data

Available space is limited using quotas.

Use *quota* command on Linux to see how many space is used and free.

Data Storage

Files in HOME are saved every hour in snapshots.

- accessible in every directory under **.snapshot/.snapshot**
- files saved in hourly.X , nightly.X, weekly.X
- very useful when you accidentally deleted or changed the some files

Older files can be restored from backups, mail to

servicedesk@rz.rwth-aachen.de

and ask if these files can be restored.

Batchsystem on Linux

Oracle Grid Engine: (formally Sun Grid Engine)

submit a batchjob:

qsub [options] batchscript

option	description
-N *name*	*name* of the job
-o *o_file*	stdout redirected to *o_file*
-e *e_file*	stderr redirected to *e_file*
-w *v*	check only parameters of the script; no job submission
-M *mailaddress*	send notification mails to *mailaddress*
-m *b|e|a|s|n*	mail at begin | end | abort | suspend | no mail
-l *resource=value*	specifies *resources* needed to execute the job
-pe *p_env nprocs*	request parallel environment *p_env* with *nprocs* processes

Batchsystem on Linux

If a resource is not requested a small default is used or the recourse is not available at all.

But the job may wait longer, if more resources are requested.

recource	description
-l *h_rt=hh:mm:ss*	time the job may take
-l *h_vmem=xxxX*	main memory per process in X = [K|M|G] (Kilo-|Mega-|Gigabyte)
-l *mtype=machine type*	request machine of type [nehalem_typ1 | harpertown | barcelona | dunnington]
-l *software=#_licenses*	*request license for ISV software #_licenses normally =1*
-l *threads=nthread*	specify number of threads the program uses
-l *threadsperslot=nthr*	bundle *nthr* threads into one slot
-l *procsperslot=nproc*	bundle *nproc* MPI processes into one slot

Batchsystem on Linux

Benchmarks: using all cores in most cases leaded to slowdown

thus: we do not schedule CPU‘s or cores, but „slots“

CPU / Node typ	Slots	UEs per Slot	Sockets | Cores | (hardware) Threads per node
nehalem_typ1	8	2	2 | 8 | 16
barcelona	8	2	4 | 16 | 16
dunnington	12	2	4 | 24 | 24
harpertown	4	2	2 | 8 | 8

it is still possible to (over)populate all Units of Execution with work

(options `threadsperslot, procsperslot`)

Batchsystem on Linux

Example batch script:

```
#$ -N Hybrid-Test-Job
#$ -l h_rt=00:15:00
#$ -l h_vmem=1500M
#$ -l threads=4
#$ -pe mpi* 5
#$ -l mtype=harpertown
$PSRC/pex/462 # build the example program a.out
cd 462
$MPIEXEC $FLAGS_MPI_BATCH a.out
```

- qsub options can be specified in a batch script
- the line must begin with magic cookie “#$“ (no spaces before)

Caution: commenting out `$VAR` may cause an error;
add a space: `# $VAR`

Batchsystem on Linux

Job management commands:

see job status:

qstat [-j job-ID] [-u USER] …

delete job:

qdel <job-ID>

start a management GUI:

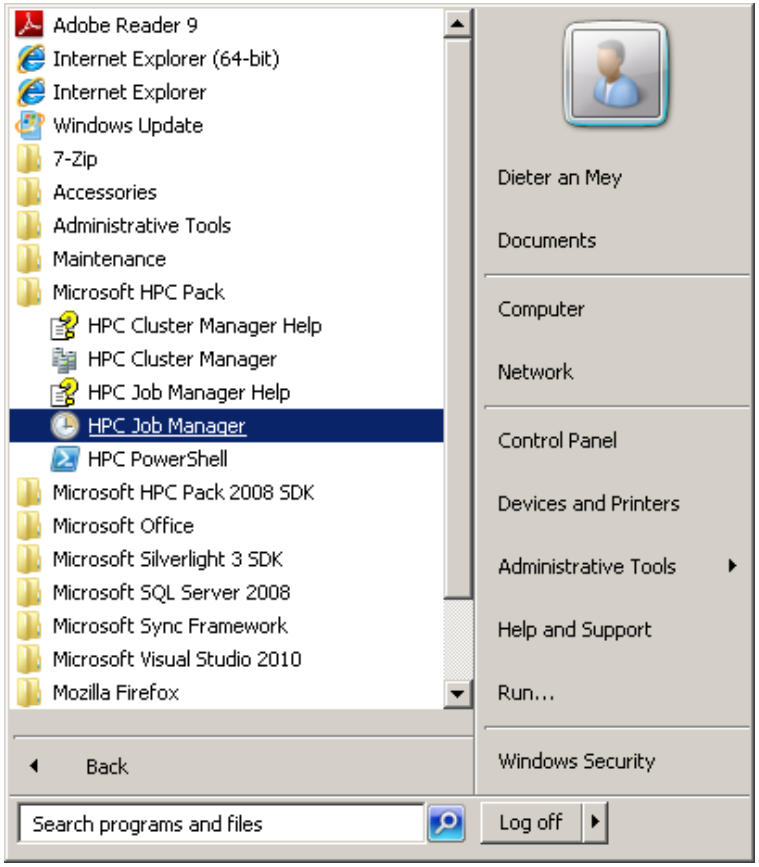
qmon

DEMO

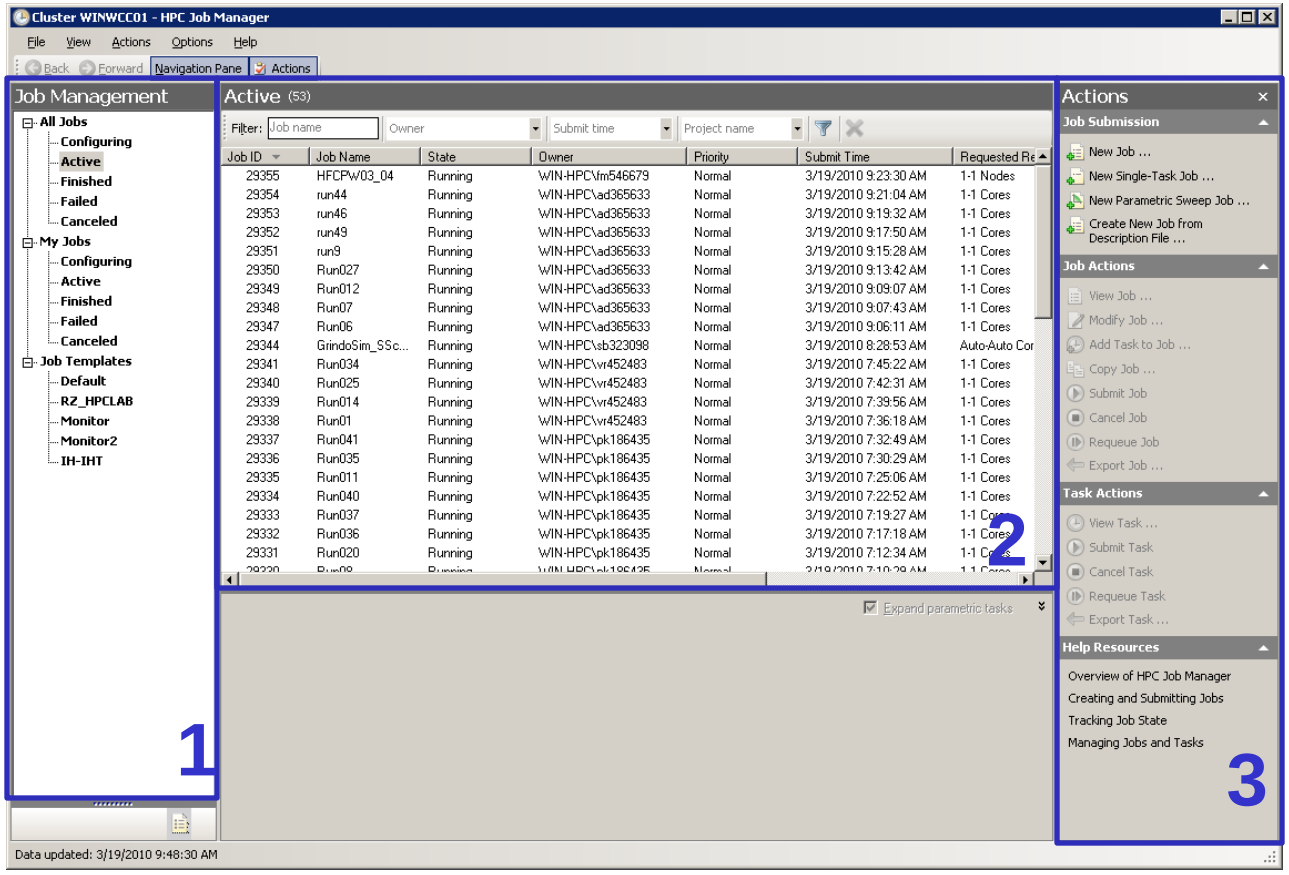
Batchsystem on Windows

HPC Job Manager contained in *Microsoft HPC Pack:*

Start → All Programs → Microsoft HPC Pack → HPC Job Manager



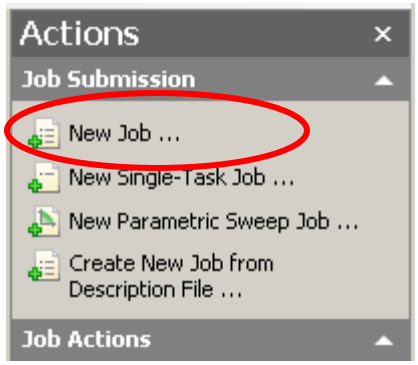
Batchsystem on Windows

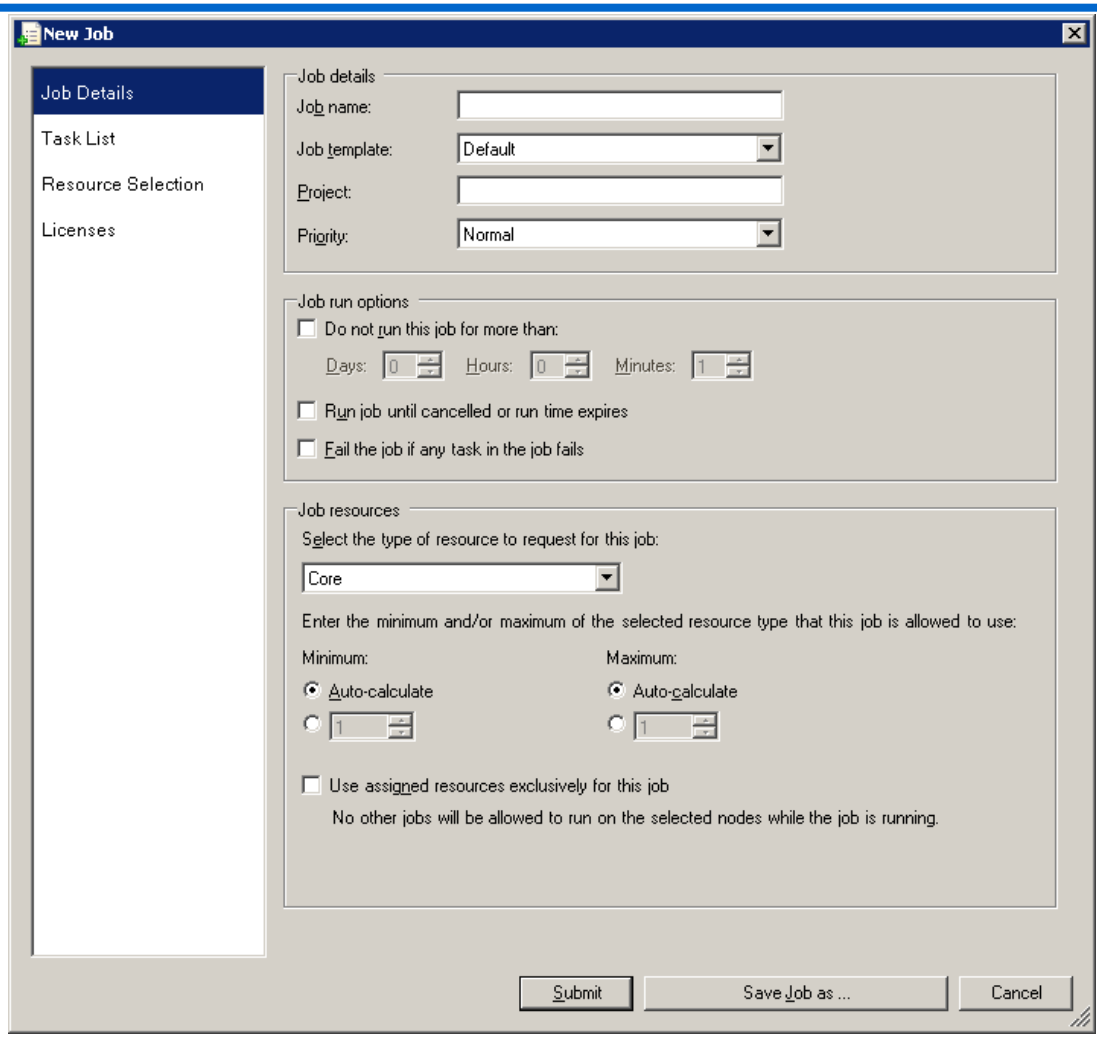


1: different preconfigured views
2: see job status
3: action pane

Batchsystem on Windows

- You are free to choose a *Job Name* and a *Project Name* as you like.
- You might specify runtime and failure options for the job.
- Resource allocation:
  - Per *Core*, or
  - Per *Socket*, or
  - Per *Node*.



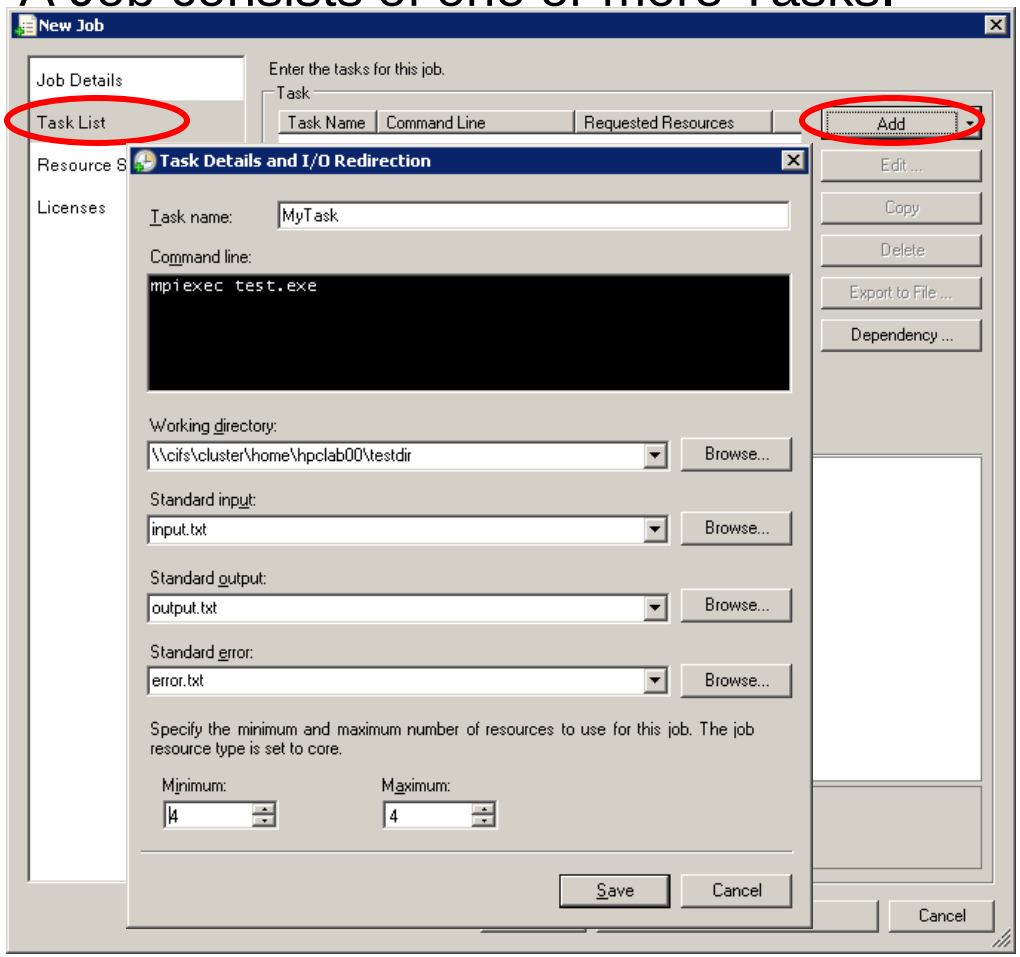


Batchsystem on Windows

- Resource Allocation Granularity:
  - Per *Core*: Get *n* processor cores. No further restrictions, for example it cannot be assumed that a (sub)set of cores shares the same main memory (→ not suited for Shared-Memory).
  - Per *Socket*: Get *n* sockets. On our cluster, currently each socket has four cores, thus it can be used for Hybrid or Shared-Memory (up to four threads per process).
  - Per *Node*: Get *n* nodes. The number of cores can be specified using the *Resource Selection*.
- If you use OpenMP: Set OMP_NUM_THREADS environment variable, otherwise you would get as many threads as there are cores.

Batchsystem on Windows

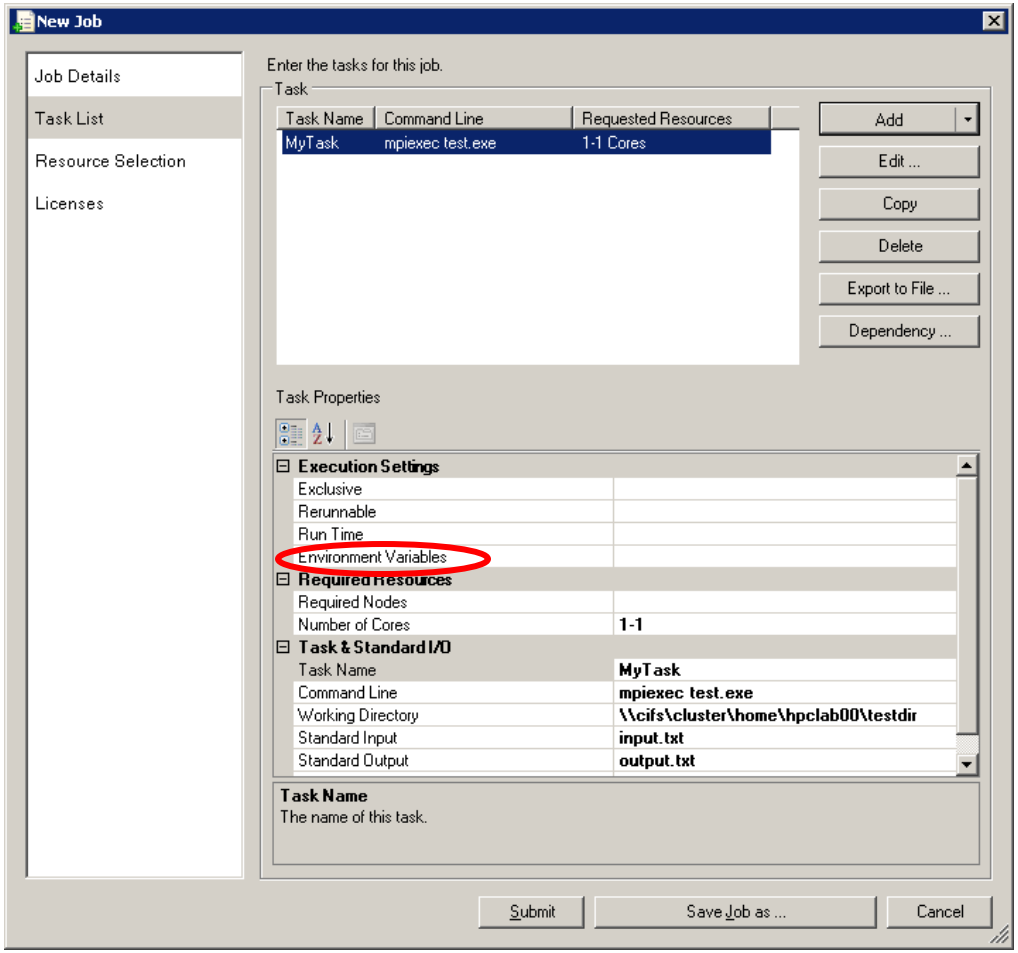
A Job consists of one or more Tasks.



- *Command Line:* You can specify the full path to a program including program options or to a `.bat` or `.cmd` file.
- You have to use network paths instead of drive letters (`\\cifs\cluster\home\...` instead of `H:`) in any path.
- For MPI Tasks just include `mpiexec` in the *Command Line*, do not specify any other MPI options.

Batchsystem on Windows

After task creation additional parameters can be set.

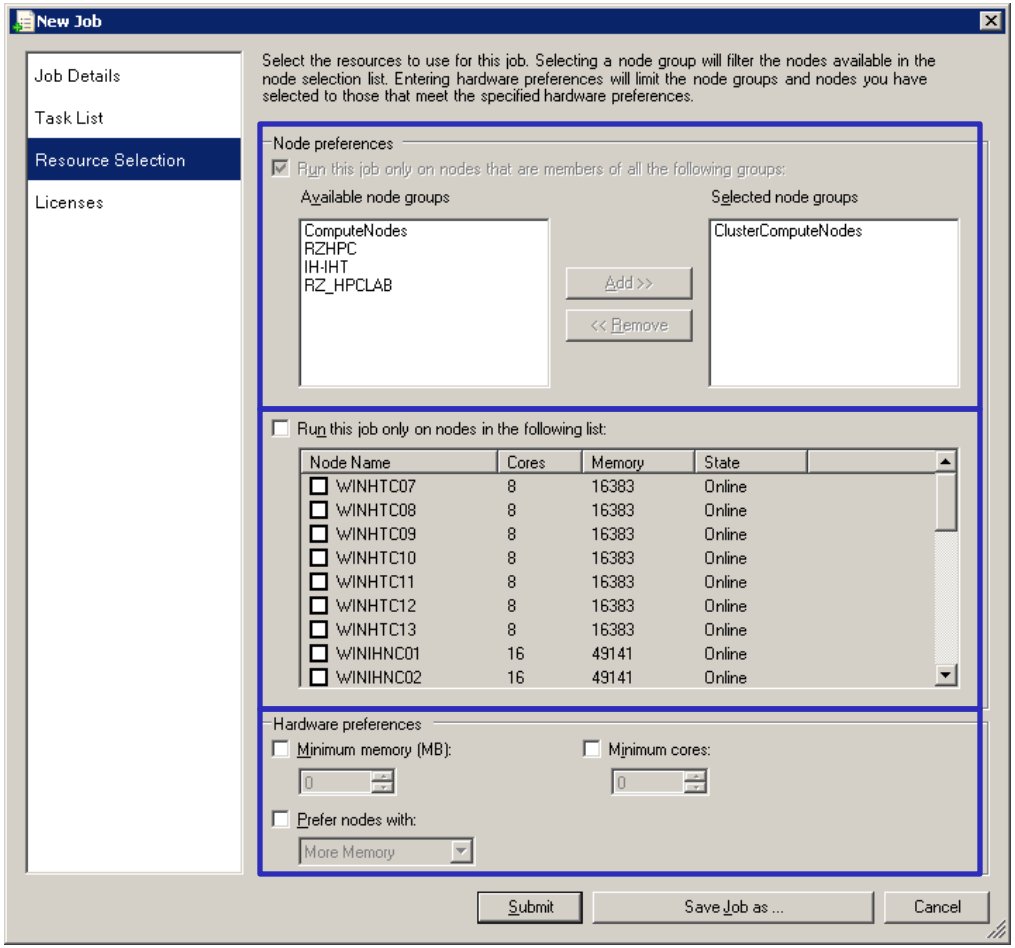


Set environment variables for serial and OpenMP jobs here.
NOTE: for MPI programs variables must be set in the command line to be set for all processes:

```
mpiexec -genv OMP_NUM_THREADS 2
```

Batchsystem on Windows

Some restrictions for the node selection can be specified.



- Allow selected classes of nodes only.
- Allow a selected set of nodes only.
- Allow nodes with enough memory or cores only.

DEMO

Any problems or questions?

1. See detailed information in our Compute Cluster User‘s Guide:
   www.rz.rwth-aachen.de/hpc/primer

2. Have look at our webpage:
   http://www.rz.rwth-aachen.de/hpc

3. Send a mail to:
   sevicedesk@rz.rwth-aachen.de

4. Ask now or during the whole workshop.

The end

Thank you for your attention!

Questions?