



Interactive European Grid

Exercise

Sven Stork
HRLS, Stuttgart

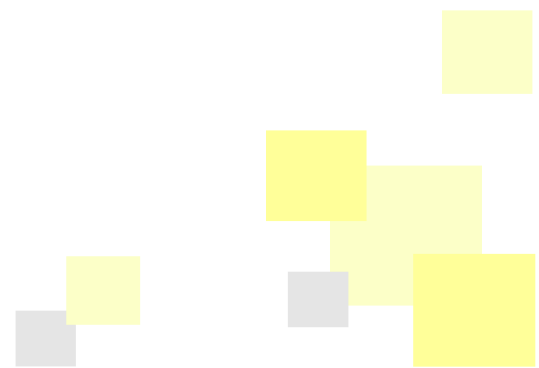
MpiGridUserCourse07, Dublin

- Part 1
 - ▶ IMB (aka Pallas)
- Part 2
 - ▶ “Interactive” /bin/sh
 - ▶ gnuchess console
 - ▶ gnuchess console 2
 - ▶ gnuchess MD

Part 1

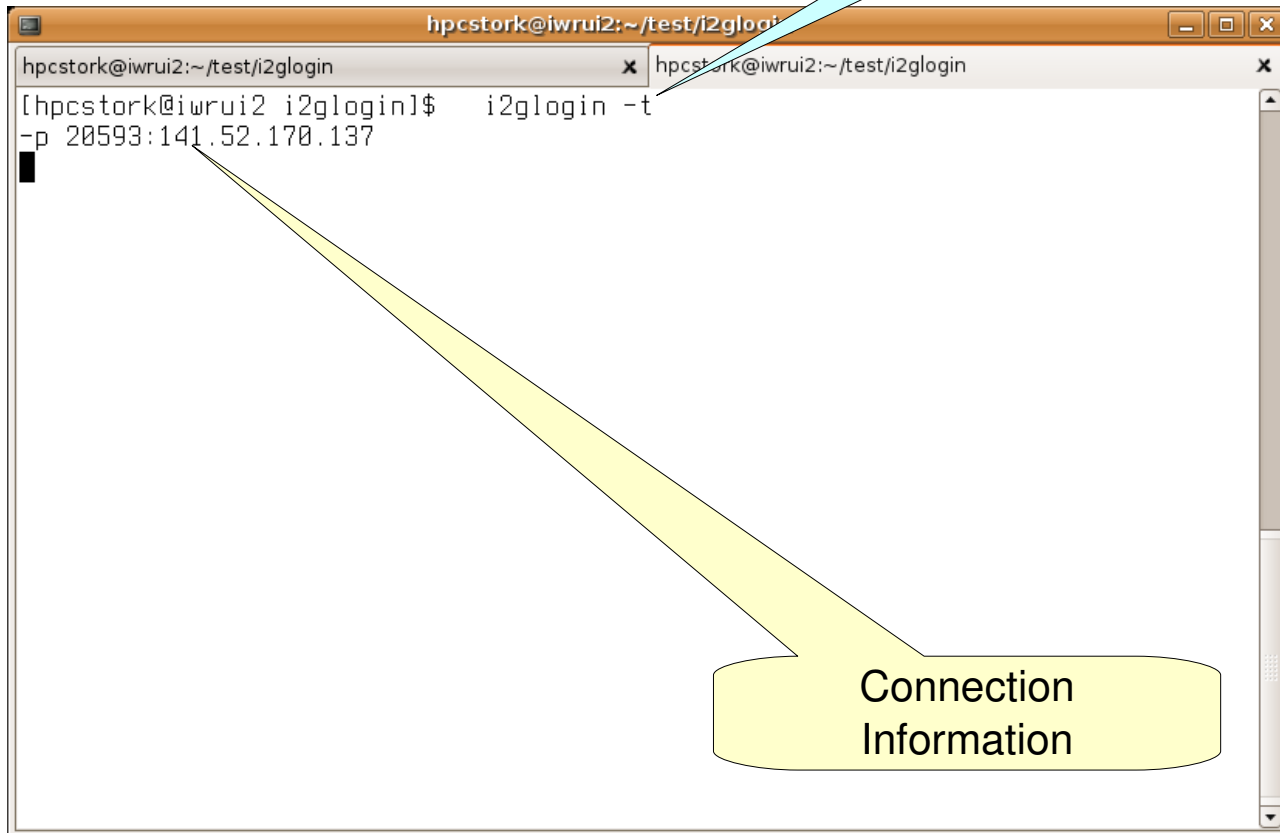
- ❑ download the IMB
 - ▶ <http://www.intel.com/cd/software/products/asmo-na/eng/219848.htm>
- ❑ compile IMB with Open MPI (default)
- ❑ create a corresponding JDL file
 - ▶ 4 processes
 - ▶ Test : pingpong, barrier
- ❑ Find the fastest cluster in the infrastructure

Part 2



“Interactive” /bin/sh

- first shell
 - ▶ start i2glogin



```
hpcstork@iwru2:~/test/i2glogin
[hpcstork@iwru2 i2glogin]$ i2glogin -t
-p 20593:141.52.170.137
```

-t : handle new line correctly

Connection Information

□ second shell

- ▶ copy i2glogin binary to local directory
- ▶ create i2glogin.jdl file
 - Executable
 - i2glogin
 - Arguments
 - contact information from shell 1
 - -r for remote connection
 - -t for handling new lines
 - -c “...” for specifying the “real” command
- ▶ submit job

“Interactive” /bin/sh

```
hpcstork@iwruui2:~/test/i2glogin
hpcstork@iwruui2:~/test/i2glogin
[hpcstork@iwruui2 i2glogin]$ cp /opt/i2g/bin/i2glogin .
[hpcstork@iwruui2 i2glogin]$ cat i2glogin.jdl
Executable      = "i2glogin";
Arguments       = "-p 20593:141.52.170.137 -t -r -c /bin/sh";
InputSandbox    = {"i2glogin"};
StdOutput       = "std.out";
StdError        = "std.err";
OutputSandbox   = {"std.out","std.err"};
[hpcstork@iwruui2 i2glogin]$ i2g-job-submit --vo imain i2glogin.jdl

Selected Virtual Organisation name (from --vo option): imain
Connecting to host i2g-rb01.lip.pt, port 7772
Logging to host i2g-rb01.lip.pt, port 9002

*****
                                JOB SUBMIT OUTCOME
The job has been successfully submitted to the Network Server.
Use i2g-job-status command to check job current status. Your job
identifier (edg_jobId) is:

- https://i2g-rb01.lip.pt:9000/e2fgQmfL9TU6WAAtLkkygg

*****

[hpcstork@iwruui2 i2glogin]$
```


“Interactive” /bin/sh

- first shell
 - ▶ wait for shell prompt
 - ▶ have fun

- ❑ download gnuchess binary
 - ▶ <http://www.hlr.de/people/stork/gnuches>
- ❑ create “run.sh” file

```
#!/bin/sh

chmod 0755 i2glogin
chmod 0755 gnuchess

./i2glogin $@ -c $PWD/gnuchess
```

□ update/create JDL file

```
Executable      = "run.sh";  
Arguments       = "-p 20593:141.52.170.137 -t -r ";  
InputSandbox   = {"i2glogin", "gnuchess", "run.sh"};  
StdOutput      = "std.out";  
StdError       = "std.err";  
OutputSandbox  = {"std.out", "std.err"};
```

□ same procedure as for the interactive shell

gnuchess console (2)

- simplify the gnuchess console program by using the I2G JDL extensions

- run the gnuchess in the MD (refer to live-demo/video)