MCNEG 2004 at NPL

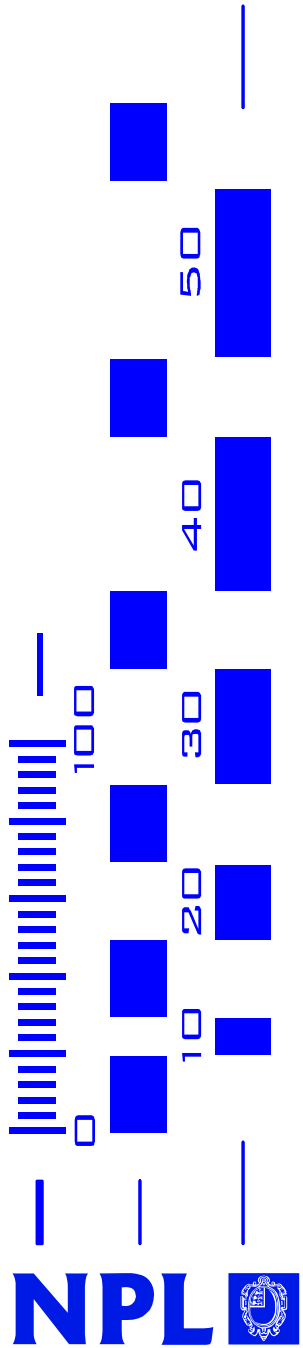
Distributing EGS on the NPL United Devices Grid

Simon Duane

National Physical Laboratory, UK

simon.duane @ npl.co.uk

16th March 2004



What's a grid?

- Like electrical power distribution
  - Producers and consumers linked in a transparent way
  - Don't need to know who burnt the oil (or where)

- Computing grid
  - Producers have cpu / disk / memory
  - Consumers have computational tasks

- Producers = donors / desktop pc owners … the victims
- Consumers = grid users … me, Hugo, et al.

Let's get together…

NPL

Some pre-history: hardware for MC at NPL

- 1987 – VAX 11/785 (finance)
- 1988 – microVAX II (dosimetry)
- 1989 – Meiko Computing Surface (dosimetry)
  - Initially 4 processors, then 28 processors
- 1990 onwards – PCs (dosimetry)
  - DOS + Pharlap 386 + Lahey F77 (not networked)
  - Win3.1, Win3.11, Win95 (networked)
  - Linux (dual boot desktops)
  - Linux (dedicated)
- 2002 up to 9 Linux boxes dedicated to MC
  - But not a proper cluster…

=> We know about parallel execution

NPL

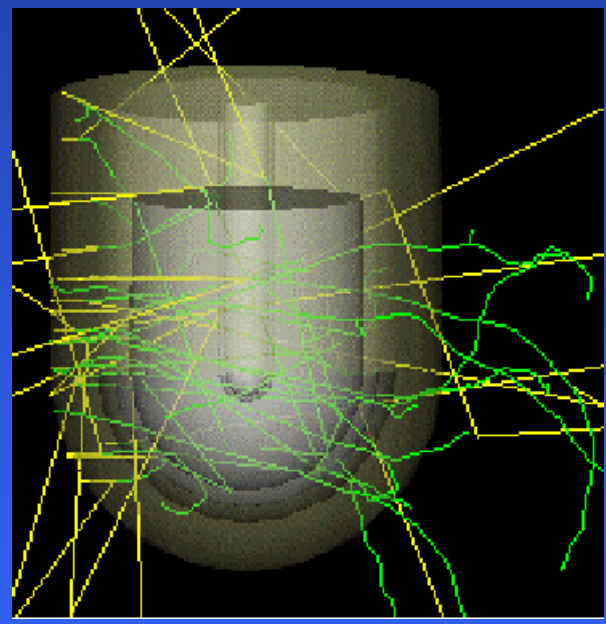
Parallelizing Monte Carlo simulations

- In principle, no problem:
  - we need billions of histories anyway
- In practice, need to
  - split task at start
    - Independent random sequences – we use Marsaglia-Zaman (easy to generate and label 10^8 sequences)
  - Merge results at end
    - Dose calculation - combine and improve statistics
    - Phase-space generation – concatenate files
- Multiple instances of the executing program are independent (they don't need to talk to one another)

NPL

output data: fixed SIZE problems (e.g. dose)

- Can always be made to work:
  - Volume of data io is fixed
  - Increase job duration to make comms/comp large enough



DOSCHAM

NPL

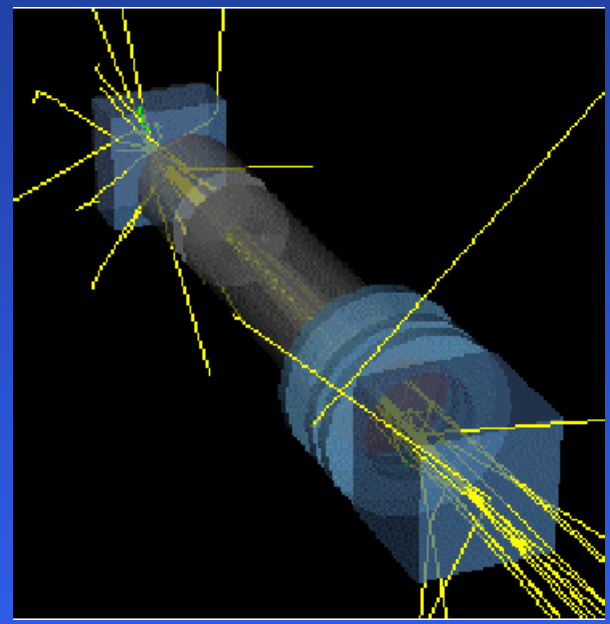
output data: fixed RATE problems (e.g. phase space)

- May or may not be worth it:
  - Electron beams – no
  - Photon beams – maybe

(for our linac simulations, anyway)

NPLLINAC and DOSCHAM:

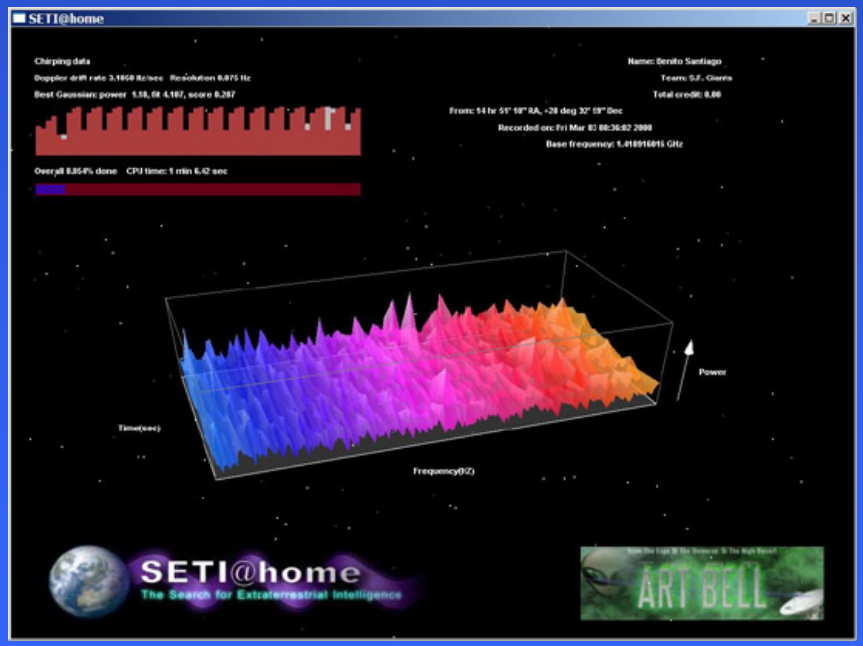
EGS4/PRESTA usercodes written by David Shipley



NPLLINAC

Grid examples

- GIMPS
  - Great Internet Mersenne Prime Search:
- Seti@home
  - You've seen the screen saver:



New Scientist - Microsoft Internet Explorer provid...

File Edit View Favorites Tools Help

Back Search Favorites Media

Address http://www.newscientist.com/news/news Go Links

The World's No.1 Science & Technology News Service

Largest prime number ever is found

15:11 02 December 03

NewScientist.com news service

A 26-year-old graduate student in the US has made mathematical history by discovering the largest known prime number.

The new number is 6,320,430 digits long. It took just over two years to find using a distributed network of more than 200,000 computers.

Michael Shafer a chemical engineering student at Michigan State University used his office computer to contribute spare processing power to the Great Internet Mersenne Prime Search (GIMPS). The project has more than 60,000 volunteers from all over the world taking part.

"I had just finished a meeting with my advisor when I saw the computer had found the new prime," Shafer says. "After a short victory dance, I called up my wife and friends involved with GIMPS to share the great news."

Prime numbers are positive integers that can only be divided by themselves and one. Mersenne primes are an especially rare type of prime that take the form 2^{p}-1, where p is also a prime number. The new number can be represented as $2^{20,996,011}$-1. It is only the 40th Mersenne prime to have ever been found.

Opening p Internet

NPL

The NPL UD grid – (i)

- The desktop machines run MS Windows NT / 2000 / XP
  - UD also allow linux, or Sun, or AIX, or any mixture…
- Linux server(s)
  - We have two, located centrally.
- Secure
  - Optional encryption – we don’t need it.
- Unobtrusive
  - Users unaware their pc being used by someone else

NPL

The NPL UD grid – (ii)

- 650 staff, approx 600 desktop PCs
  - Mostly 3 year life, so reasonably current models
  - Mostly idle, most of the time
- All networked, using managed switches
  - 1Gbs (backbone – not sure exactly where)
  - 100 Mbs (new building)
  - 10Mbs (old buildings)
- PCs at NPL are configured with a standard disk image
  - Includes UD Grid agent software (since late 2003)
  - >200 PCs have agent installed

NPL

The NPL UD grid – (iii)

- 2 linux servers (could have been one)
  - Filestore (DB2 database)
  - Grid management services (5)
    - Poll server
    - Dispatch server
    - Realm server
    - RPC server
    - File server
- 100 agent licenses (with option for more…)
  - Dispatch server limits number of devices that can be actively running jobs.

(show slide from UD training)

NPL

What happens?

- Each device runs agent as service
  - i.e. on boot, independent of user login to Win2k
- Every 2 mins:
  - Agent:

  “hello – I’m waiting for something to do”
  - Poll Server:

  “thanks – do this” [sends program and data files]

  or
  - Agent:

  “hello – I’m waiting for something to do”
  - Poll Server:

  “thanks – there’s nothing on at the moment – come back a bit later”

NPL

Setting up and running code on the grid

- Start with fortran source in CygWin (copy from linux?)
- Make all io to current directory (if not already)
- Implement split and merge
- Compile (e.g. g77 in CygWin)
- Copy to another directory
  - The executable, cygwin1.dll, any input data files, the UD loader.exe
- Build a UD program module and data packages
  - persistent and workunit datas
- Run in the testagent (on local PC)
- Upload to UD server and launch job(s)

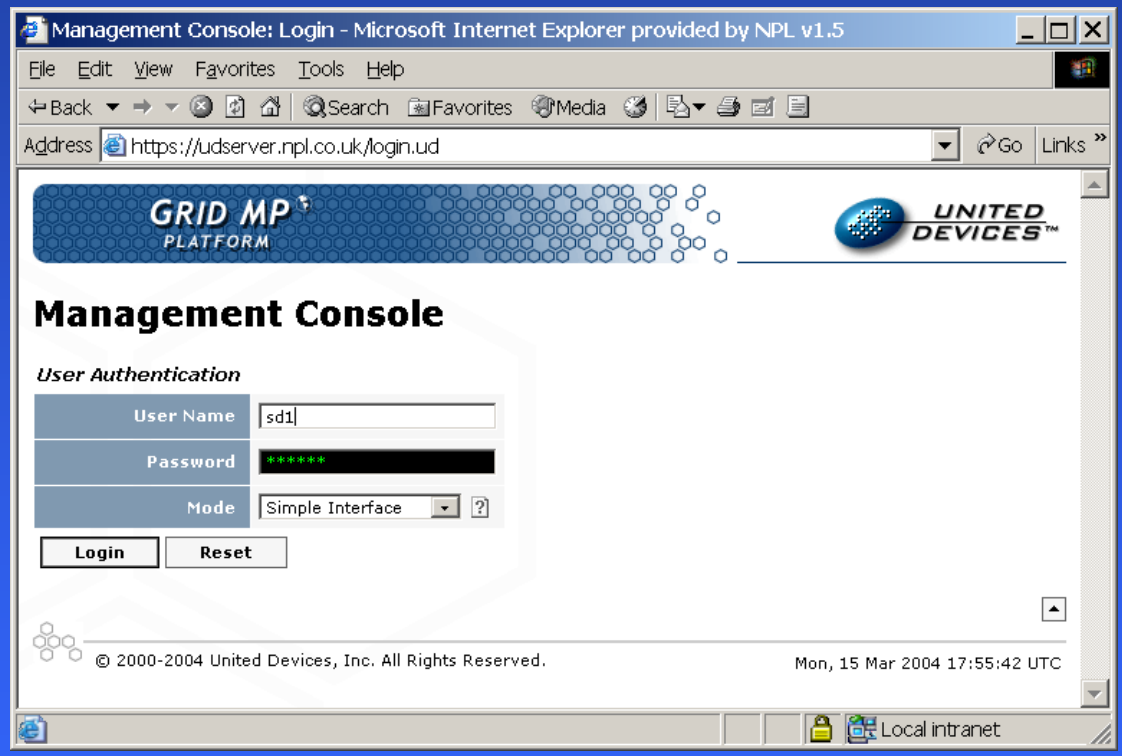
NPL

Grid user interface - options

- Web-based
  - Interactive
  - User-friendly
  - Good for getting to know what's there
- Use XML-RPC or SOAP
  - Programmable – C++, Java, etc
  - Scriptable – perl, python, etc

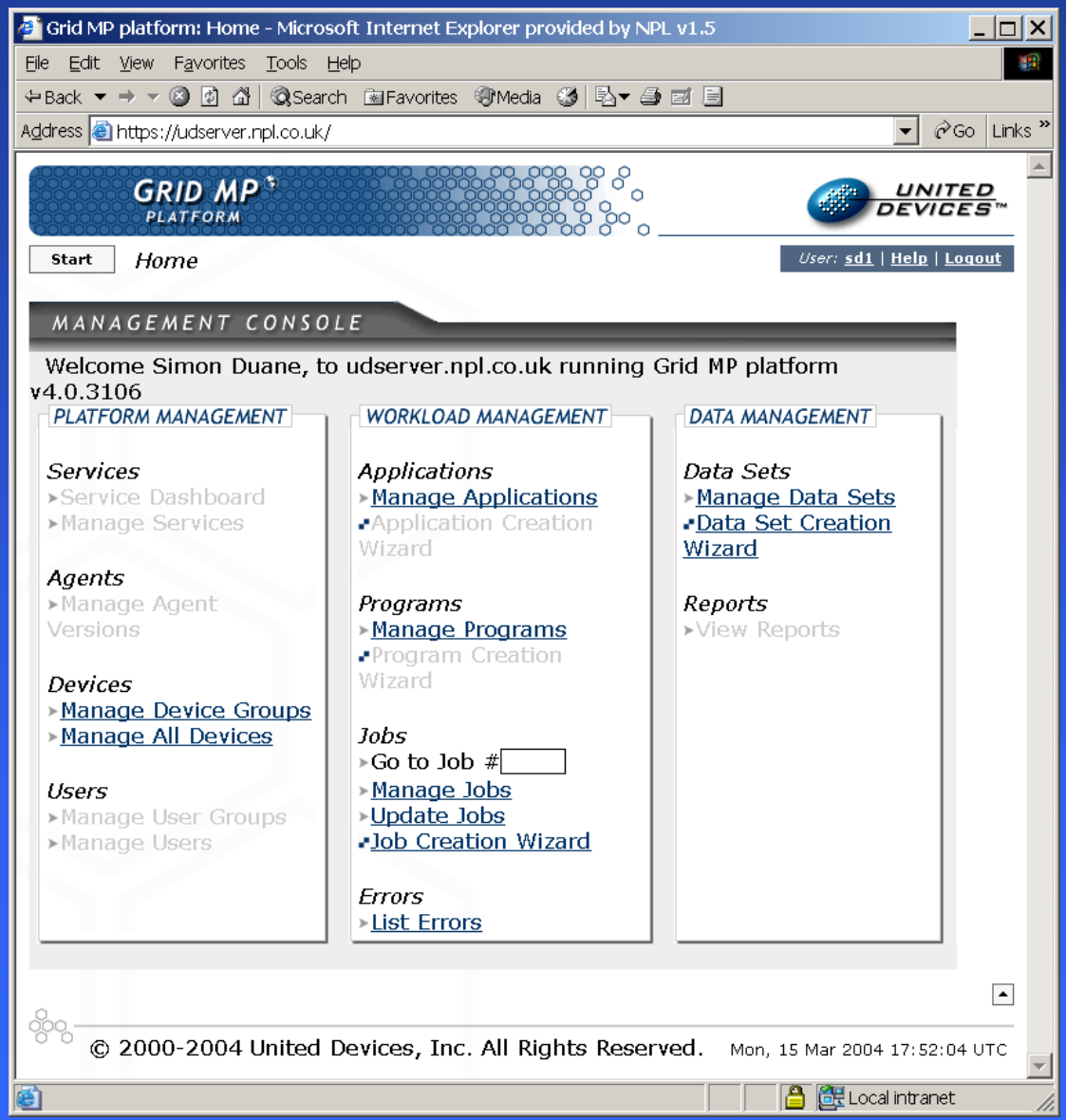
NPL

Web based – login



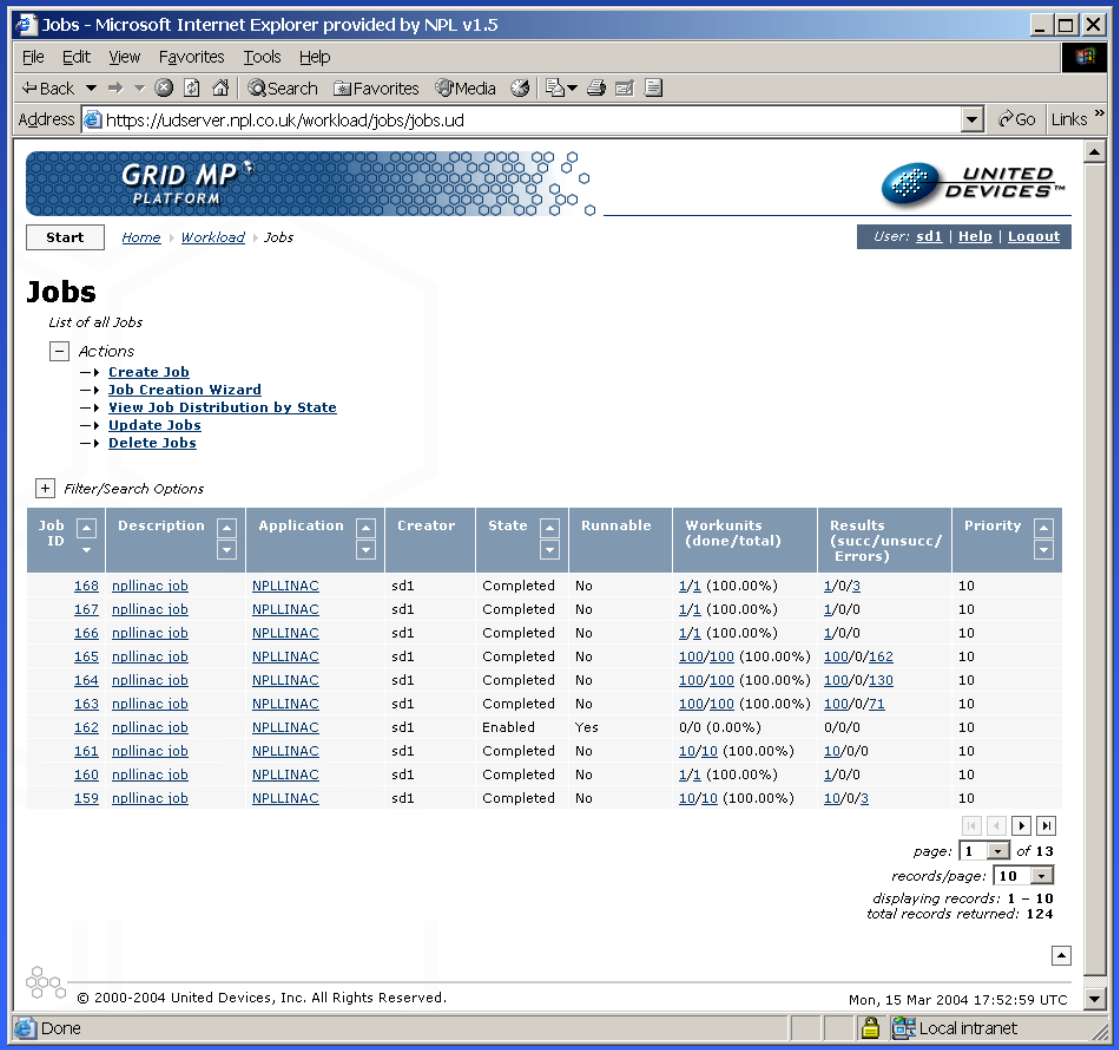
NPL

Web based – console



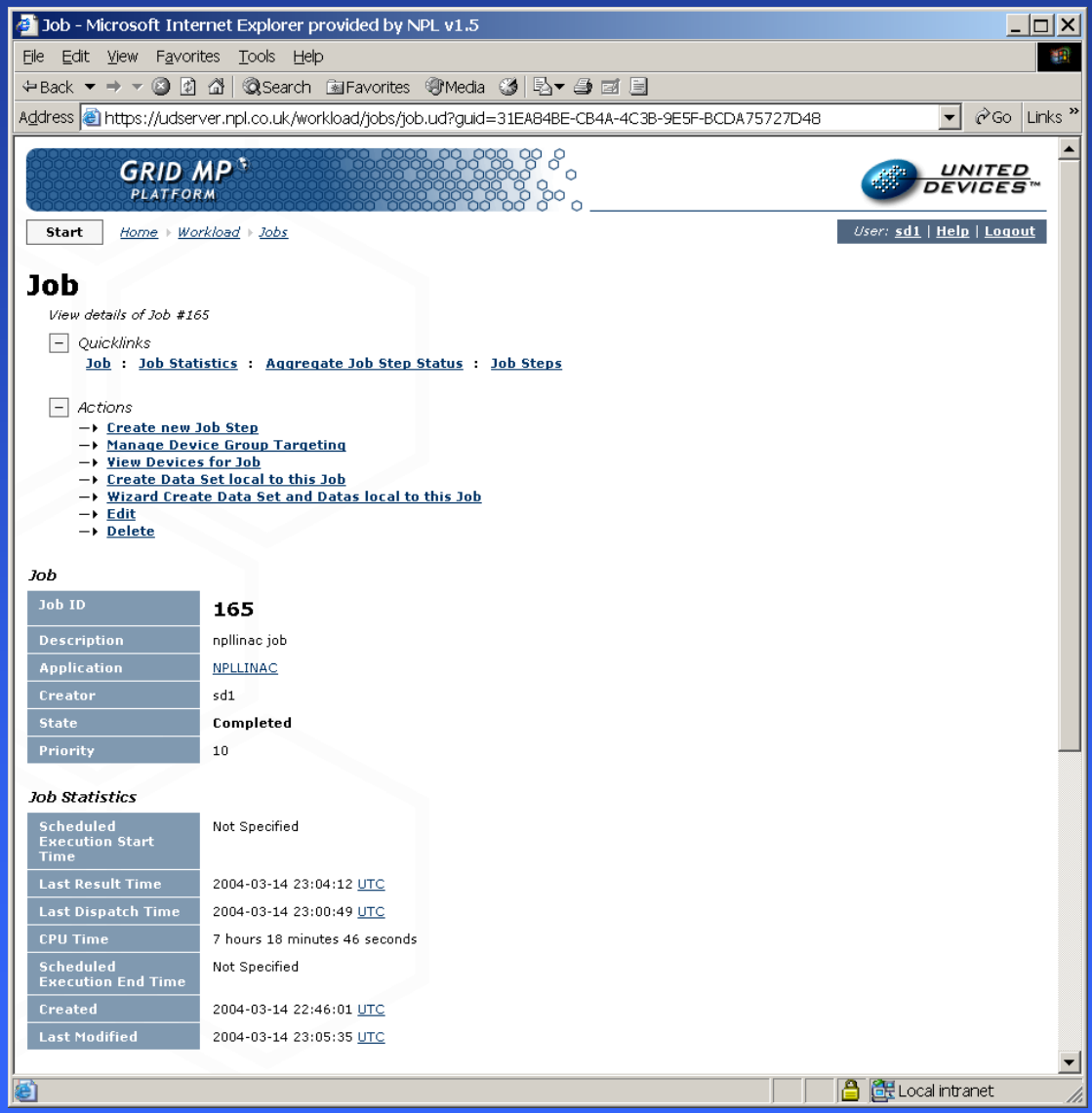
NPL

Web based – manage jobs



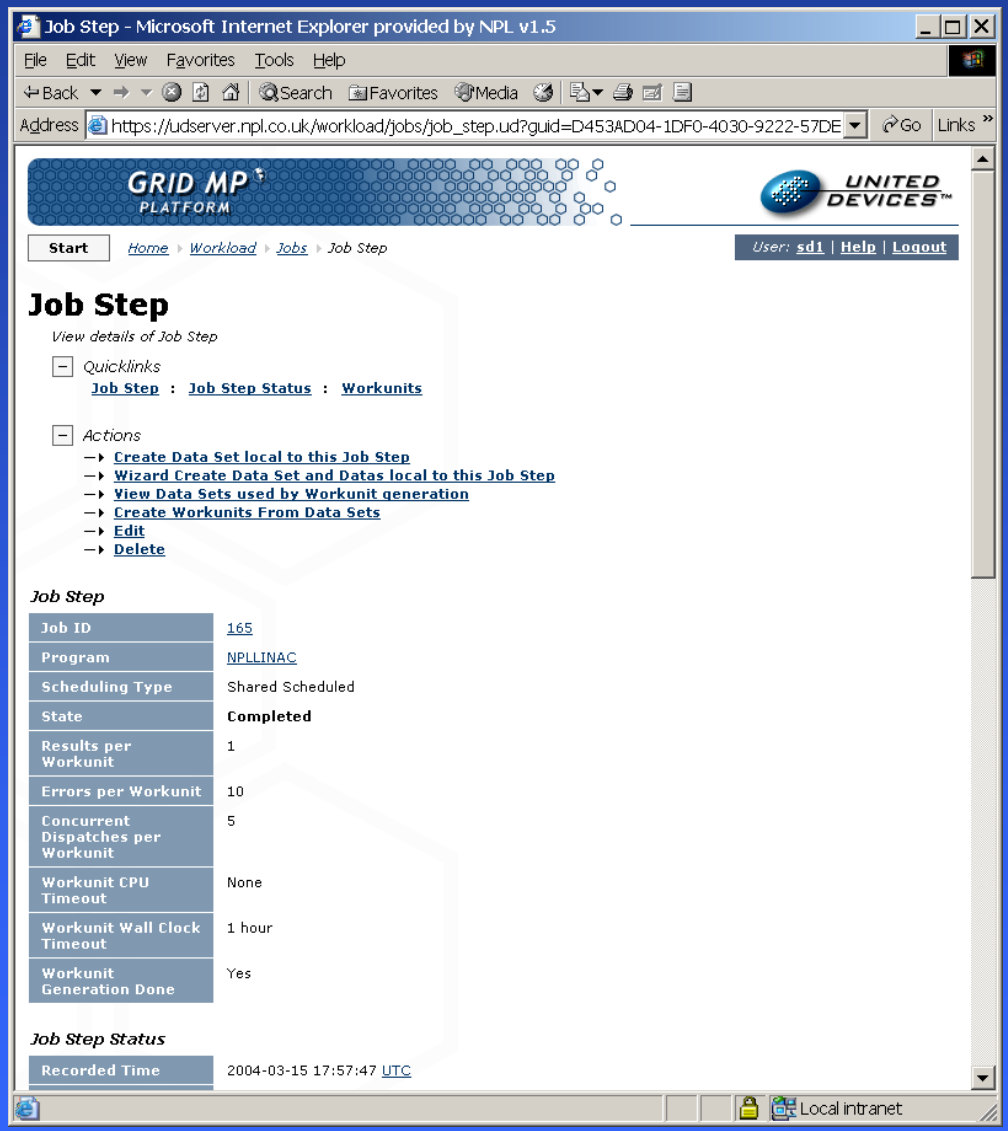
NPL

Web based – (after select job id)



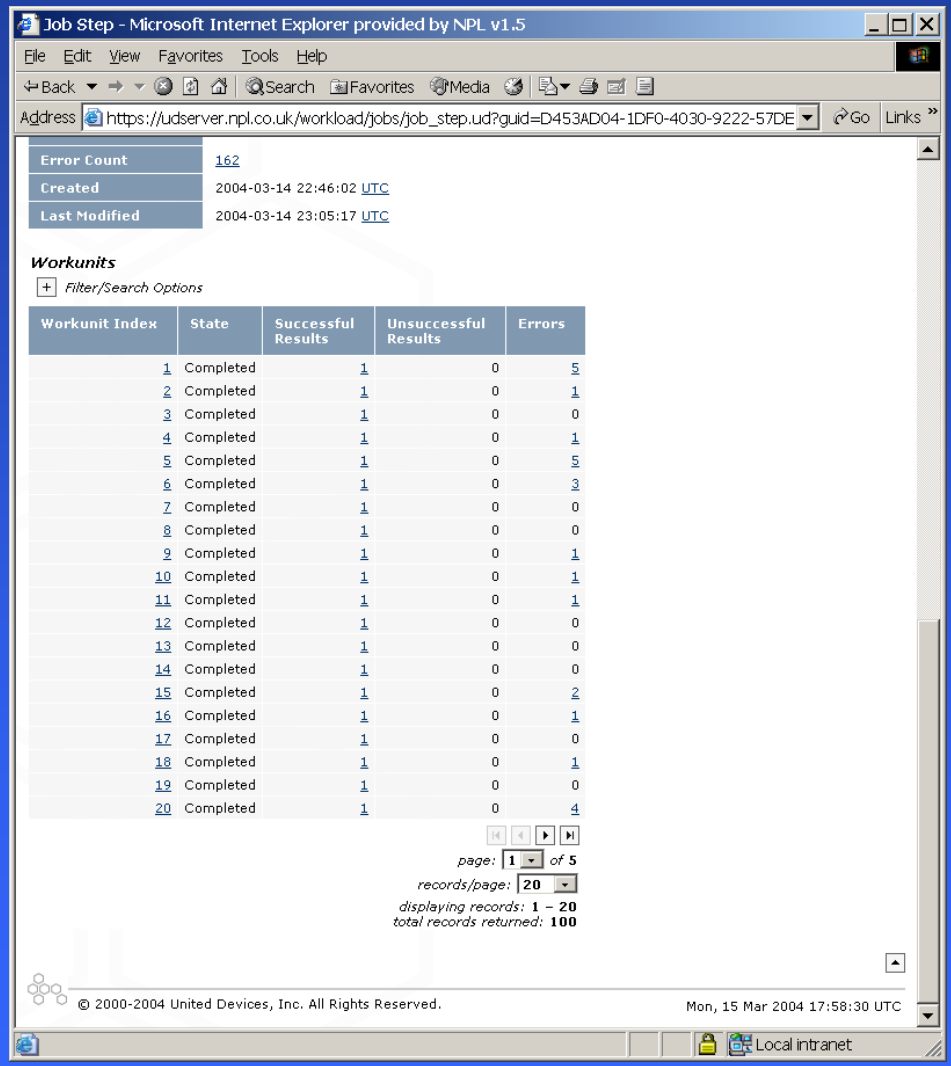
NPL

Web based – (after select job step)



NPL

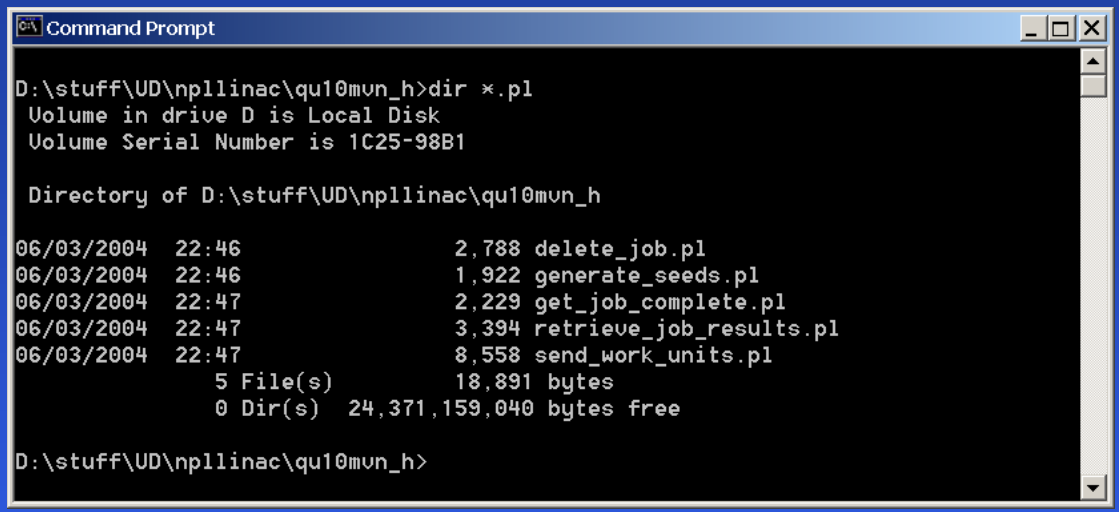
Web based – browsing results

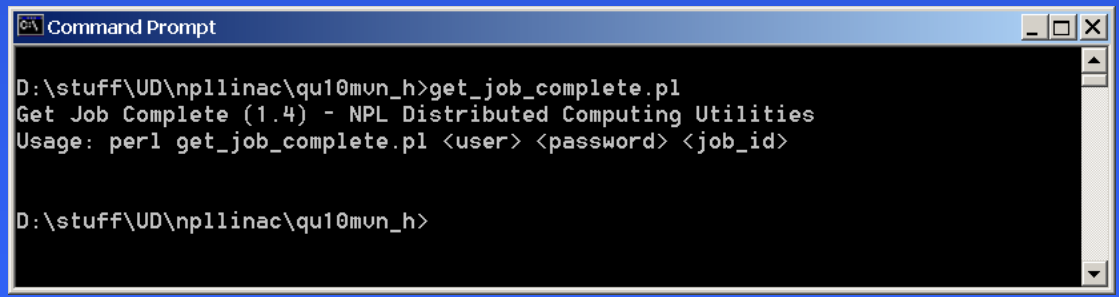


and so on

NPL

Or use scripts – e.g. perl

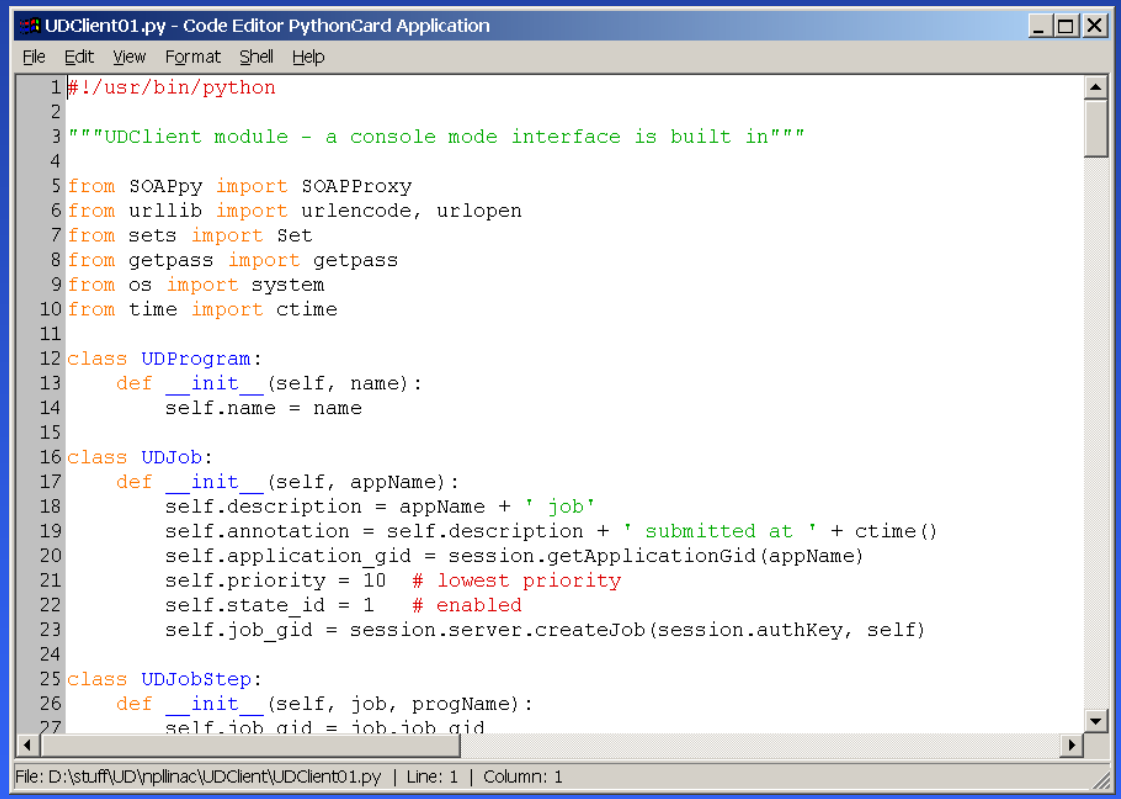




Perl scripts at NPL adapted by Keith Lawrence from UD examples

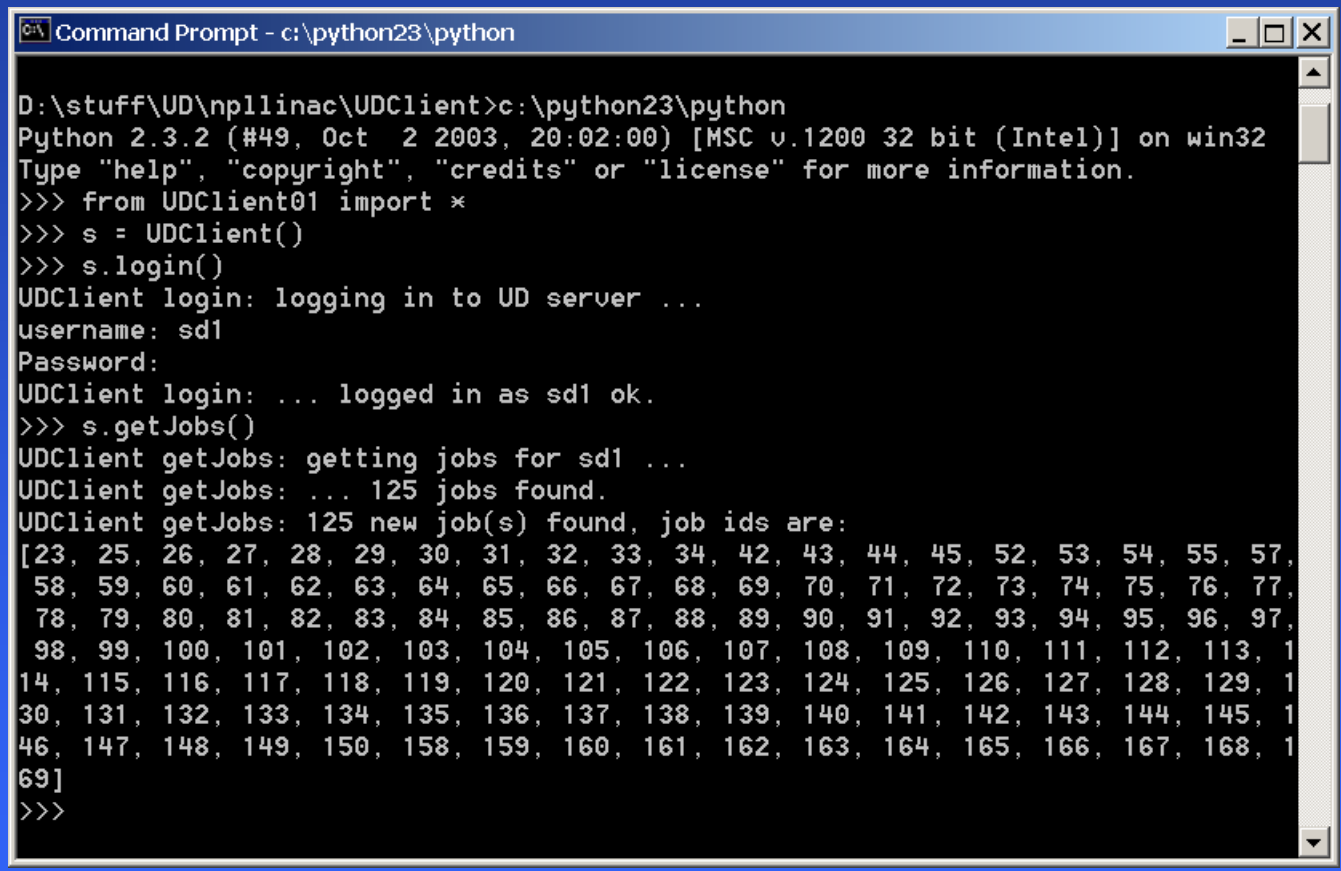
NPL

scripts – e.g. python



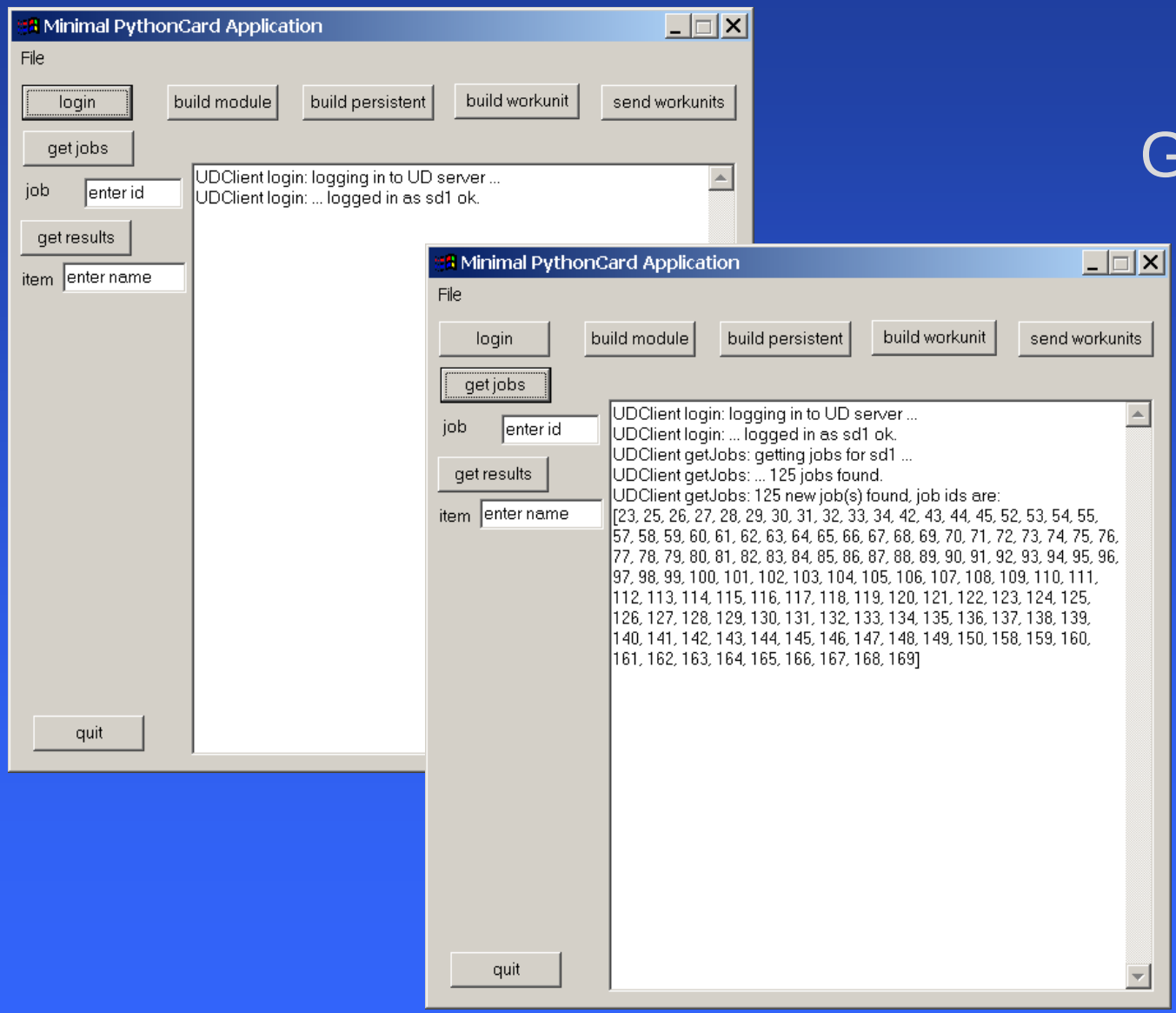
source code

python in use – command line



NPL

python in use – GUI



GUI

NPL

Early results

- PTRAN
  - Proton transport code (Hugo Palmans' talk…)
  - Physics (ion chambers in phantom in proton beam)
  - Jobs submited Friday evening, ready Sunday lunchtime (equivalent to about 4 months on a desktop PC)

- EGS4/PRESTA – NPLLINAC usercode
  - Phase space generation
  - A benchmarking exercise (so far)
  - Aim to discover the (io) limits of the system

NPL

NPLLINAC – performance on a laptop

- 4, 6, 8, 10, 12, 16, 19 MeV electrons onto a
  - Tungsten target
  - With or without Al filter
  - With collimator (makes a beam 11cm diameter at 125cm
  - Range rejection, brems splitting turned on …

	4MV heavy filt	19 MV light filt
Histories /sec	4870	1025
Particle yield	0.005	0.23
Data rate	700 byte/sec	6.4 kbyte/sec

Run on a Pentium M 1.6GHz (g77 in CygWin)

NPL

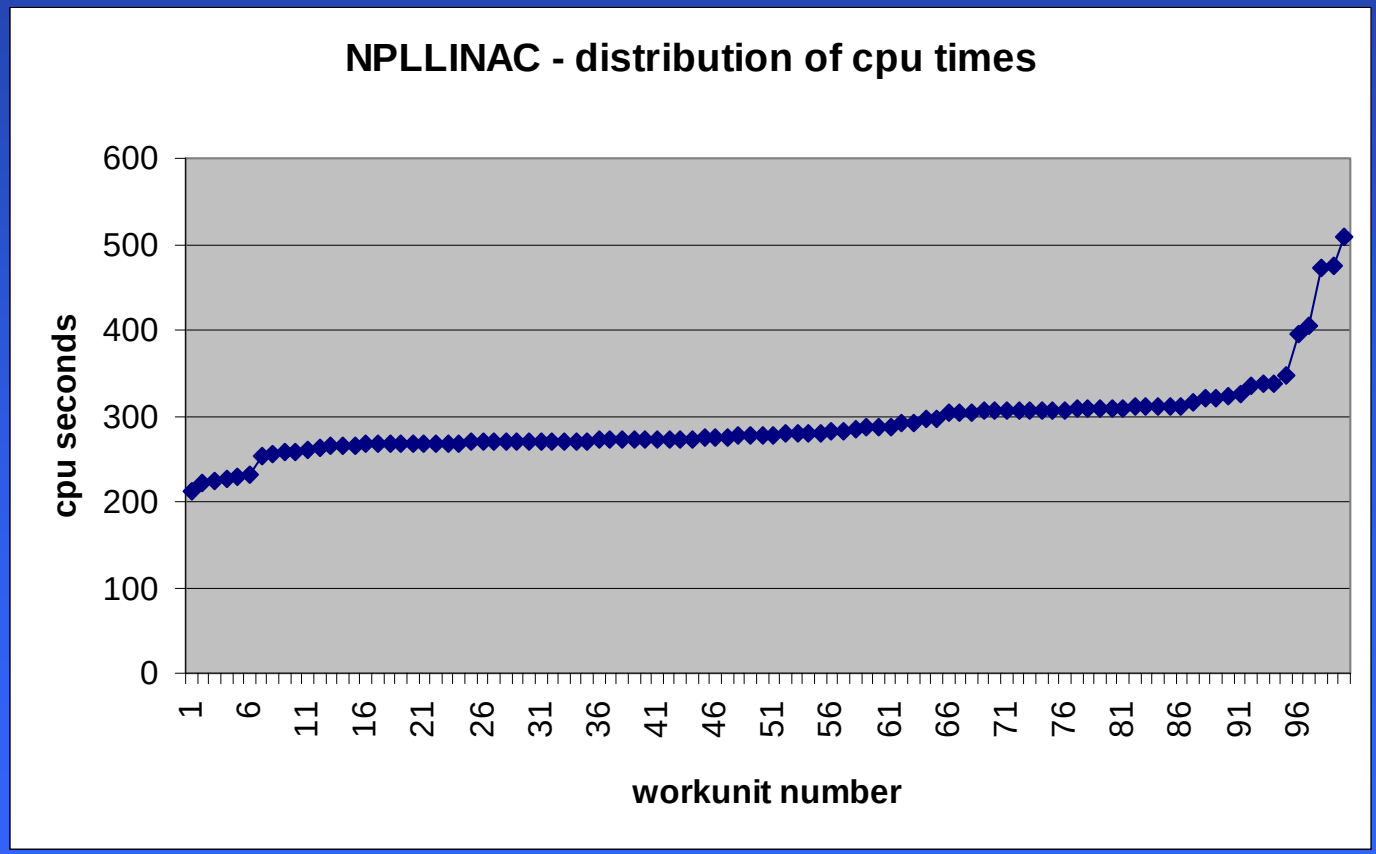
Grid of 100 devices?

- 4MV heavy filtration
  - 100 x 700 byte/s = 70 kB/s
  - Should be manageable
- Even 19MV light filtration should be ok …

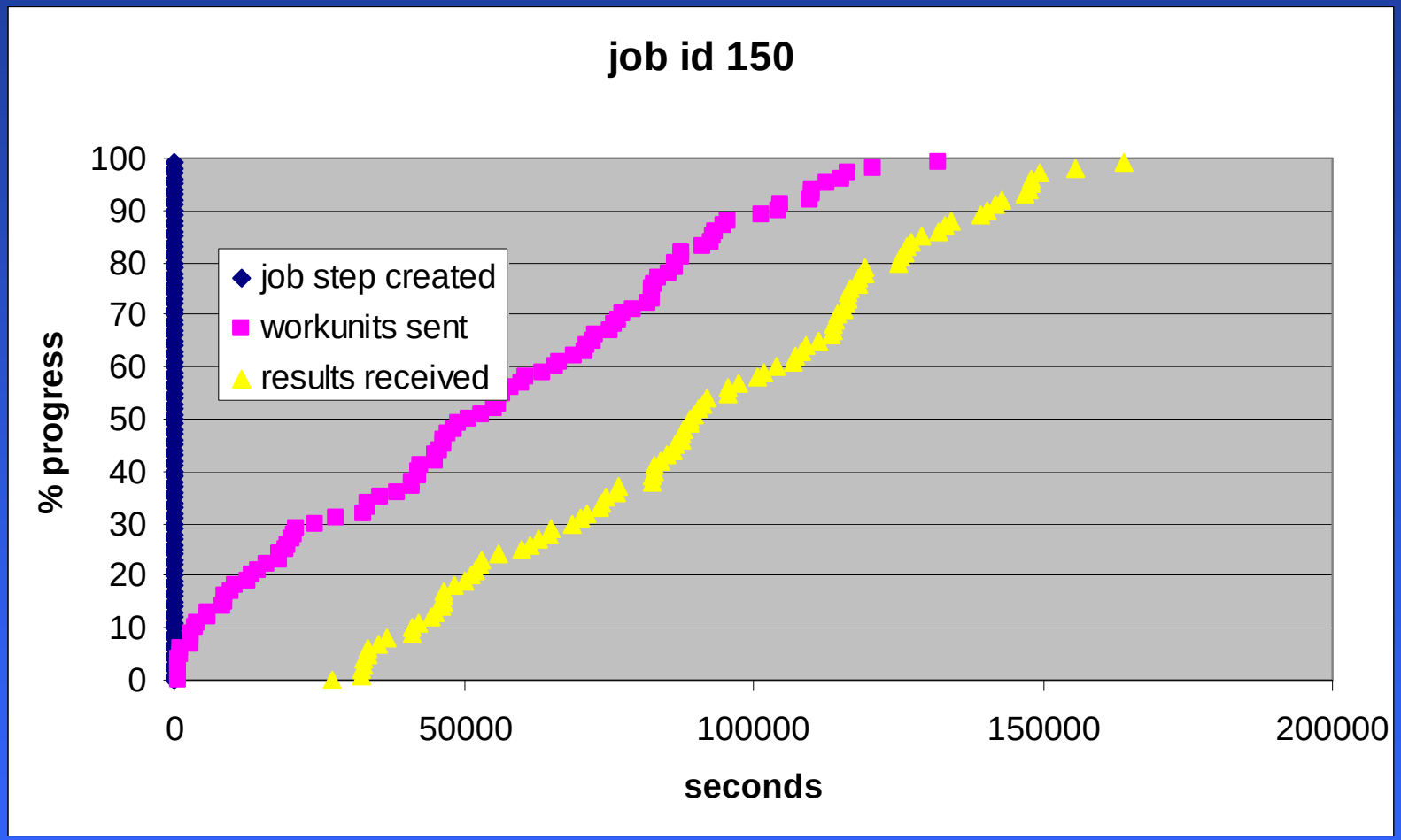
NPL

Some results

- How long does each agent spend on a workunit?



What about progress?



NPL

Efficiency?

- Most realistic measure:

  total cpu time / elapsed time / number of devices available
- Result (really preliminary, no tuning of server)

  About 50% for jobs that last 4 minutes (NPLLINAC)

  About 90% for jobs that last 10 hours (PTRAN)

NPL

Conclusions (i)

- Grid computing really *is* here now
- It is available on Windows as well as linux, etc.
  - (it's not that I am a Windows enthusiast, but that's what's 95% of the cpu on site run)
- It makes obvious economic sense:
  - Licence is ~ $100 per device
    - Transferable when PC hardware is replaced
    - Transferable when switch from MS Windows to Linux
    - The business requires that most staff have PCs – the marginal cost of harnessing all those cpus is not much more than the electricity cost of leaving them on out of hours
    - United Devices have to compete with a lot of Free Software (on Unix if not on Windows) – their pricing is "flexible".

NPL

Conclusions (ii)

- There are some unresolved issues in our system – maybe because many/most devices have the same speed (100 Mbs) network interface as the servers
  - Workunit result error rate is sometimes high
    - (recalculated automatically)
    - Devices may be trying to hit server with 30 Gbyte/min…
  - Devices can timeout during initial transfer from server
    - (resent automatically)
    - Server may try to send out 100 x 20MB simultaneously

NPL

Ever the optimist

- Our use of the Grid has only just begun
  - Already good for production runs with fixed data size code
- I expect that with a bit of learning / tuning
  - It will even be good for short “steering” runs, of a few minutes (though probably not a few seconds)

NPL

Post-script (i)

- What about coprocessors?
- In 1988, on hearing about the Weitek chip that would offer 5 Mflops, BLIF said

  That could change the way I work

- In 2004, I googled for weitek and found out about
  - www.clearspeed.com
  - Their CS301 coprocessor:
    - 64 processors on a chip
    - 2 FPU per processor
    - 12800 Mips
    - 25.6 Gflops
    - 2W at 200MHz
    - 10 Gflops / Watt
- That could change the way *I* work…

NPL

Post-script (ii)

- Who are ClearSpeed?
  - I googled some more, and found an “ex-Inmos employees re-united” site, and found that
    - some had gone on to Meiko (late ’80s)
    - Others had ended up at ClearSpeed
- The Transputer lives on, in spirit …

NPL