

# Data Migration

**Ian J. Bush**

Computational Science and Engineering Department

CCLRC Daresbury Laboratory

Warrington WA4 4AD

I.J.Bush@dl.ac.uk



The CSAR service will cease to be on 30<sup>th</sup> June 2006

- It's best to assume that after that date your data will NOT be available

So if you want to continue using it the data will have to be migrated to somewhere you can get at it

- Home
- HPCx

and you have roughly one month to do it ...

# What did that last line say??

---

Note the last line of the previous slide:

**“and YOU have roughly one month to do it ...”**

It is you, the user, whose responsibility it is to transfer the data that you require. It is just not feasible for us to do it

- Filesystem and tape incompatibilities
- Shear volume of data – what do you want to keep ?
- Aggregate bandwidth
- You may have to convert the data format

## GET TRANSFERRING NOW

There's only a finite amount on bandwidth available, roughly enough to transfer 64 TBytes if the maximum we have measured is sustained throughout the month. This may sound a lot, but there IS a lot.

If you don't start transferring data soon you may lose it!

It's going to take longer than you think!

---

It may be that you can't just transfer your files immediately, or all at once

- You might have to retrieve them from tape at CSAR which takes time
- You may have a number of very large files, only a few of which you can store on disk at once
  - tape at CSAR -> disk at CSAR -> disk at HPCx -> tape at HPCx?
- You may have to convert the endianness
  - Most easily done at CSAR

**GET TRANSFERRING NOW**

The most tricky part of getting your files ready for HPCx is that Newton is a *little endian* machine, while HPCx and the Origins are *big endian*. The nett result of this is that

## Binary Files Are Incompatible Between Newton and HPCx

and you will have to convert them. This is much, much more easily done on Newton than on HPCx, and yet another reason to

## GET TRANSFERRING NOW

Consider how one might hold the four byte hexadecimal number 0x12345678 in memory. There are a number of orders that can be used, the two most common being

- *0x12, 0x34, 0x56, 0x78*
  - Big endian - used by IBM (HPCx), most Sun systems and also on the Origins
- *0x78, 0x56, 0x34, 0x12*
  - Little endian - used on Newton, most Linux clusters

As the order is different, binary files generated by Newton are incompatible with HPCx

For Fortran I/O the easiest way to convert is to use environment variables to change the format in which you want to output your data. You can then write a little program that reads in the data in the Itanium's usual form, then writes it out in a form suitable for HPCx.

For example, the following bash/ksh command will set Fortran (8.1 and 9.0 compilers) unit number 60 to output data in a form that can be read on IBMs (including HPCx).

```
export FORT_CONVERT60=ibm
```

If you now run a code all data associated with unit number 60 will be interpreted as being read or written for an IBM. There are other options which are discussed fully in the man pages.

If you have a binary file written by C or C++  
equivalent environment variables **DO NOT EXIST**

- You will have to write a conversion program

It is difficult, if not impossible, for us to provide a generic solution to this problem as what will be required will depend upon the structure of the file, i.e. what data is in the file (are the next 8 bytes a double, or two floats, or 8 chars or what?)

Basically binary files are inherently unportable

- But quick to write

If portability is a problem for you you should consider using one of the portable data formats, such as NetCDF or XDR, or at the very least have a conversion program available.

I'm sure you all know most of this already but the way to transfer your files is via some variant of scp.

Remember it is like message passing! Transfer in large chunks. Use the tar command to create one file from many, and gzip the resulting tar file to cut down on the amount you need to transfer

On CSAR:

```
$tar cvf my_stuff.tar dir1 dir2 ...
```

```
$gzip my_stuff.tar
```

...

On HPCx:

```
$gunzip my_stuff.tar.gz
```

```
$tar xvf my_stuff.tar
```

Probably the two most common ways to transfer files between CSAR and HPCx are sftp (secure file transfer protocol) and scp (secure copy). N.B., neither service supports ftp or rcp, and you shouldn't use them!

I prefer sftp:

```
$sftp ijb@login.hpcx.ac.uk
```

And then follow the prompts

Scp I find less easy, but is better for scripts and such like

```
$scp ~ijb/stuff \
```

```
ijb@login.hpcx.ac.uk: /hpcx/home/z001/z001/ijb/stuff
```

An alternative to scp and sftp is the gridftp facility:

```
globus-url-copy -vb -p 8 \  
gsiftp://login.hpcx.ac.uk/hpcx/home/z001/z001/chrisj/work/bin2G \  
file:///home/ngs0330/bin2G
```

The interesting thing is the `-p` flag, this specifies how many parallel data streams will be used. Potentially this can improve performance.

Table 1: Bandwidths measured when transferring files of 1-2 Gbyte between the three machines considered in this report.

Source location	Destination	transfer method	No. parallel data connections	Transfer Bandwidth (Mbytes/s)
Edinburgh	HPCx	gridftp	1	2
Edinburgh	HPCx	scp	1	2
HPCx	Edinburgh	gridftp	1	3
HPCx	Edinburgh	scp	1	2
HPCx	Manchester	gridftp	1	11
Manchester	HPCx	gridftp	1	20
HPCx	Manchester	scp	1	10
Manchester	HPCx	scp	1	12
HPCx	Manchester	gridftp	1	10
HPCx	Manchester	gridftp	2	22
HPCx	Manchester	gridftp	4	35
HPCx	Manchester	gridftp	6	47
HPCx	Manchester	gridftp	8	37
HPCx	Manchester	gridftp	16	37
Manchester	HPCx	gridftp	1	19
Manchester	HPCx	gridftp	2	22
Manchester	HPCx	gridftp	4	18
Manchester	HPCx	gridftp	6	26
Manchester	HPCx	gridftp	8	13
Manchester	HPCx	gridftp	16	11

Chris Johnson and Stephen Booth have written a technical report on this, see:

[http://www.hpcx.ac.uk/research/hpc/technical\\_reports/HPCxTR0603.pdf](http://www.hpcx.ac.uk/research/hpc/technical_reports/HPCxTR0603.pdf)

Also on the CSAR web site:

[http://www.csar.cfs.ac.uk/user\\_information/service\\_end/index.shtml](http://www.csar.cfs.ac.uk/user_information/service_end/index.shtml)

- Time is short – get transferring now
  - Binary files generated by Newton are not compatible with HPCx, and will need conversion which is most easily done on Newton
  - Remember to use tar and gzip to keep transfer times down
  - The most common way to transfer files is scp or sftp
  - Consider using gridftp which may be faster
- But, most importantly

# GET TRANSFERRING NOW