

University of Warwick Site Report

Ben Morgan
Tom Latham

THE UNIVERSITY OF
WARWICK

1: A Bit of History...

- Particle Physics group (EPP) established in Department of Physics in early 2004.
 - 3 Academics, 4 Postdocs, 2 PhD Students by end of 2004
- Group has expanded over the years, now standing at
 - 5 Academics, 6 Postdocs, 10 PhD Students.
 - Working on LHCb, T2K, Neutrino Factory/MICE and Detector Development.
- Employs 1 postdoc (Ben Morgan) as a Physicist Programmer to support group's computing needs.
 - Tom Latham providing increasing support in Grid/LHC and generic HEP computing.

2: Computing Setup

- Our core desktop system is provided by Warwick's Centre for Scientific Computing (CSC)
 - An interdisciplinary group which maintains a research and training environment employing high-performance computing to solve research goals.
- System based on openSUSE 11.0 with a large range of scientific software and tools including compilers, numerical libraries and MPI.
- Home and data storage provided on NFS from Warwick ITS SAN system.
 - 1.1T home area for EPP
 - 3.1T data storage for EPP
- CSC workstations $O(200)$ form a Cluster of Workstations (COW)
 - Serial and parallel jobs submitted through PBS batch system.
 - Jobs all nice'd so (in general!) do not interfere with desktop usage.

3: Services

- Web services provided by Warwick ITS SiteBuilder system
 - Content management with WYSIWYG and raw HTML editors
 - Provides extra tools for integrating forms (with online payment) and other tools.
- Mail services provided by Warwick ITS
 - Original Groupwise system replaced with Exchange – much improved service!
 - Old Unix mail system still available, but very limited quota.
- Backups provided by Warwick ITS for /home and /storage areas
 - Offsite tape system with recovery from end of previous 2 weeks and 2 months.
 - We have requested two (limited) recoveries so far and has worked well.
- Windows services (XP, thinclient) provided by core ITS system.

4: Choice of Setup

- CSC system was chosen at group startup due to a lack of resources:
 - Space and money in department
 - Time – best employment for a Physicist Programmer is research!
- We continually review our computing provision
 - CSC still the best from resources aspect.
 - We get batch computing 'for free' using the COW - with some limitations.
- However, we have experienced problems with HEP software on non-SL system
 - openSUSE compilers typically more recent than SL
 - Some 'standard' HEP packages missing (e.g. AFS, though that is now resolved!)
 - Binary compatibility when source code not available.

5: Desktop Workstations

- Specification is provided by CSC based on compatibility tests with their system.
 - Armari (2004)
 - Viglen (2005-2007)
 - Dell (2007-present)
- Intel motherboards with Pentium (now Core2) processors and Nvidia graphics
 - Very happy with Dell Optiplex workstations – reliable and quiet.
- Total of 24 workstations with 10 x86_64, the rest i686.
 - PXE boot with automated install and updates.
- We get 3yr next-day maintenance with all systems, and service them ourselves for an extra 2yr.

6: Laptops

- Laptops purchased on an 'as needed' basis
 - User can select from University purchase scheme (Toshiba and Dell), Apple Academic purchase, or any other specific model they want.
- We don't provide administration for any laptop
 - Tom and I provide help and advice as needed.
- Remote logins to any CSC machine are possible.
- An NX server is also provided for remote desktop access.
 - Has proved useful for student projects – nxclient installed on Warwick Windows XP.

7: Software Management

- At the system level, CSC will, with restrictions on server type packages, install any package for which an RPM exists in the mainline openSUSE repositories.
- EPP are provided with a directory into which we can install HEP specific software.
- Use GNU Environment Modules for configuration of user's environment.
 - A (not so) new tool for package configuration.
 - *“The Environment Modules package provides for the dynamic modification of a user's environment via modulefiles”*
- A modulefile is a simple Tcl script that works under nearly all shells, and even within Python and Perl.
 - Administrator only has to deal with ONE file in ONE language!

8: Environment Module Use

- From the user's perspective, there's a clean and logical interface:

```
[user@i686_box ]$ module load root/5.22.00
[user@i686_box ]$ which root
/warwick/epp/2008.0/root/5.22.00/i686/gnu/4.3.1/bin/root
[user@i686_box ]$ module switch root root/5.23.02
[user@i686_box ]$ which root
/warwick/epp/2008.0/root/5.23.02/i686/gnu/4.3.1/bin/root
```

- The user has more information available on installed packages:
 - `module help <modulefile>` : provides information on the package
 - `module avail` : lists the modulefiles available to the user.
 - `module list` : lists the modulefiles loaded in the current session.

9: Environment Module Use

- Through an extra Tcl convenience library, Modules use is independent of system architecture and distribution:

```
[user@i686_box ]$ module load root/5.22.00
```

```
[user@i686_box ]$ which root
```

```
/warwick/epp/opensuse_11.0/root/5.22.00/i686/gnu/4.3.1/bin/root
```

```
[user@i686_box ]$ ssh ubuntu_box
```

```
[user@ubuntu_box ]$ module load root/5.22.00
```

```
[user@ubuntu_box ]$ which root
```

```
/elsewhere/ubuntu_8.04/root/5.22.00/x86_64/gnu/4.2.4/bin/root
```

- User can select a package for a specific compiler and architecture if needed:

```
[user@i686box ]$ module load root/5.22.00-pgi-7.2-x86_64
```

```
[user@i686_box ]$ which root
```

```
/warwick/epp/opensuse_11.0/root/5.22.00/x86_64/pgi/7.2/bin/root
```

10: Environment Module Features

- Modules prevents two versions of the same package being loaded in same session
 - Prevents environment mangling and resultant user problems!
- Modules partially deals with inter-package dependencies
 - e.g. module load geant4/9.2 will automatically load clhep/2.0.3.3
- However, it won't resolve dependencies at unload
 - e.g. module unload clhep WON'T automatically remove geant4
 - Looking at adding functionality to deal with this.
- ***Overall, a very neat and tidy system for managing the environment for packages compared with shell scripts.***

11: Experiment Specific Issues

- Running a distribution other than SL has created challenges for experimental software.
- T2K has actually gone pretty well
 - We provide an openSUSE 11.0 'buildbot' machine to the collaboration which quickly identifies issues building from source.
- MICE has a few teething issues
 - Some incompatible binaries distributed with code, but seems solvable.
- LHCb has been the biggest, and as yet unresolved challenge
 - Installation involves download of SL binaries – seems incompatible with openSUSE.
 - Little documentation on build from source, though LHCb have been helpful.
 - Current “solution” is a local SL server for Warwick LHCb users.

12: Grid Computing

- At present our main Grid user is LHCb, though T2K and MICE are picking up.
 - Major requirement at present is the client tools.
- Installing client tools has been a bit of a challenge
 - Limited, and rather unclear, documentation for gLite.
 - Why isn't the gLite source code (clearly) available?
 - Have 'installed' gLite's binary tarball, but needed to copy some SL system libraries into its lib directory to get things to work.
 - Possibly lots of superfluous things installed as well?
- We are looking at more involvement with Grid computing though!
 - A lot to learn here about exactly what's needed and if Warwick can contribute.

13: Summary and Future Plans

- Are we the only HEP group in the country
 - Not directly administering our desktop system?
 - Not using SL??!
- We're continuing to use the CSC system as our best computing solution, but we continually review the provision.
- We are beginning to run into issues with Grid and Experiment software due to binary compatibility issues.
- We are looking at more involvement in Grid computing.