My file

SCIENCE AND ENGINEERING RESEARCH COUNCIL
RUTHERFORD APPLETON LABORATORY

INFORMATICS DIVISION

SOFTWARE ENGINEERING GROUP NOTE 21

UK UNIX USER GROUP MEETING

1/2/85

Issued by
D R Gibson

5 February 1985

DISTRIBUTION:

A D B Cox
A J J Dick
D A Duce
E V C Fielding
D C Findley
D R Gibson
P M Hedlund
G P Jones
F M Russell
C P Wadsworth
R W Witty
Common Base Group
UNIX Group

Introduction

The second UK UNIX* Users' Group Meeting was marked by a one day technical conference, held at the Kelvin Conference Centre of Glasgow University. The meeting did not seem to be as well attended as first one at Westfield College, (see DIC Note 938) possibly due to the venue. Apparently the person who asked for a venue in the North failed to attend!

The Conference Talks

Introduction and Welcome.

Sunil Das (City University)

As Chairman, Sunil told us of the new timetable, since Sheila Roberts of Bristol University had been unable to attend, and also of the intention of holding the third meeting of the UKUUG at City University on 16th December.

EUUG Paris Meeting.

Dave Tilbrook (Imperial Software)

Having organised three meetings for the UNIX community, Dave gave details of the forthcoming Paris extravaganza in April. No expense has been spared, with speakers from Bell Labs. (but not AT&T!) and even simultaneous

*UNIX is a Trademark of Bell Laboratories.

translation into other languages for the sessions, and back to English for the French and German Speakers.

York Implementation of X25 under UNIX.
Steve Smith (York University)

Steve took over the X25 development at York when Keith Ruttle left the project. Currently there are over 100 cpus licenced to run the York software, which does not include Perqs. The software provides terminal access, file transfer and mail. There are also plans to run uucp over X25, and also for job transfer and manipulation. (This should have been described by Sheila Roberts, who could not attend. As I had used the York software, it may have proved interesting.)

He described the protocol layer model (i.e. the Rainbow Books) and how the software realises the various protocols. However, the software doe impose some restrictions on the 1976 Packet level of the X25 standard, and also on some of the Coloured Book recommendations. He then went on to describe the interface between the host high level routines, and the lower levels provided in the Falcon front end processor. Communication between the two is by means of the York InterProcessor Protocol. The YIPP is simple, with no error correction, no explicit flow control, is host and front end processor demand synchronised, and allows 16 software channels to be multiplexed onto one physical line.

Steve then discussed the host network terminal handling, how each X29/TS29 call maps onto a YIPP channel, which require special kernel routines. The sequence of events needed to instigate a terminal session was outlined. Actual network access is made using a suite of routines provided in a netio library, which hides the host/fep protocols, and network i/o from the user.

Steve said that the software was very reliable, although various claims from the floor gainsaid this. One comment was "Our system has crashed once in three months - what are we doing wrong?" which seemed to sum up the feelings on reliability! Currently, there is work on making the software far more modular than it is at present, as well as work on tuning the performance. Up until now, all of the effort has gone into producing a working product, and now that that goal has been reached it is hoped to improve the efficiency.

Multiple UNIBUSes, UDA-50s and System V2.
Nigel Martin (Instruction Set)

This started out with a brief talk on versions of UNIX as supplied. System V2 was highlighted as being a particularly unhelpful system to install, since it only has drivers for two sorts of disk, and if you tell the installation boot program that you do not have either it bales out. The talk then degenerated into a sales pitch for NUXI, a port version of System V2 which seems to contain drivers for all the devices known to man, or DEC anyway! Unfortunately, only available as a binary system.

The Implementation of Rob Pike's Layers Code for Bit Map Displays.
Simon Kenyon (Prime Computers)

Rob Pike of Bell Labs. saw the Perq and decided that he could do better. After being advised by the AT&T lawyers that he could not call his new terminal the Jerq, the Blit appeared, now known as the Teletype 5620 Simon showed a video of the Blit in operation, and some of the associated software to utilise it. Unfortunately, the Blit costs about 6K dollars, the software is designed for System V, and you have to buy all of the associated software, a total of about 25K dollars.

After consultation with Rob Pike, Simon set about produce the bit map handling software, based on the way the Blit does it. By providing simple structures to represent the windows, simple routines could be used to display the windows. Actually, each window is represented as a linked list of structures, each structure contains a bit map and also indicators to show whether that section of a window is obscured, etc. The basic method involves displaying the window on the screen if it is not obscured, otherwise recursively dividing the window into sub-windows and trying to display them.

Simon said that he would post the sources for what he had done so far onto Usenet. The most incredible part came when he was asked what terminals he had used the routines on. As Simon has only just moved to Prime, and this is not an official project, he did not have access to a graphics terminal, so he uses a plotting routine which dumps the "screen" area of memory onto a 300 dot per inch line printer for displaying (and debugging) windows!

UNIX News and Mail for ESPRIT.

Alex Diediew (GEC Hirst Research Labs)

Several major European companies are collaborating to produce the system, the main ones being Bull, Siemens, Olivetti, GEC, and ICL as well as some university contributors. The idea is to pool R&D work to compete with the USA and Japan. The services will eventually include message passing, document/file transfer, as well as mail and news, and is based, initially anyway, on uucp. This is to aid development, and software distribution and portability.

UKNET/EUNET.

Lee McGloughlin (Westfield College)

As there are due to be major changes and development of the UKNET in the near future, Lee did not feel he could talk on the subject at the moment. He mentioned that UKNET has had to start charging for use since the demise of the free connection to vax135. This works out at 3p per Kbyte of European traffic, and 73p per Kbyte for traffic to the USA.

Shortly a new naming scheme will be applied to Usenet in general, to bring it into line with the ARPANET convention. The existing method of naming, i.e. "a!b!c" (user c at b routed through a) will move to "c@b.gb.uucp" in the not too distant future. Each domain can be sub-divided, so there is no reason why there could not be a "drg@rlvc.ral.gb.uucp" style of use. If traffic is limited to a given domain, any following domains may be omitted e.g. "drg@rlvc.ral". In order that sites with binary only systems will still be able to use the facilities, an interim format allows "a!c@b.gb.uucp". There was quite a heated discussion about the move from the existing address style, especially over who would control the address registration, and over possible confusion with the JANET naming scheme, e.g. "c@uk.b".

A UNIX-based System for Software Configuration Management.

Michael Hughes (Paisley College)

The Project Development Environment (PDE) produced at Paisley allows long file names (even under V7), project defined attributes, multiple versions, projects and project hierachies, and controlled access. By providing an alternative C library, and relinking some utilities, the file names that the user sees are mapped into unique ones, using attributes defined for each object, project and user. This means that each user sees the same set of names as every other user, although the mapping of the unique names to user names may mean that the users see different versions of the files. By associating different access attributes to different users and projects, it is possible to control the access to sets of files, and also to do some checking on developments, e.g. any changes to files may be automatically logged.

Persistent Programming Project.

Michael Atkinson (Glasgow University)

Strongly typed languages have existed for quite a while, and it has been noted that in the majority of cases, a strongly typed language aids the debugging of software. However, strong typing only exists at execution time for most of these languages, so any data stored beyond the running of the program has to be explicitly written, and then explicitly read again at the next run of the program. For those systems which have a large amount of long lived data, it may take a lot of software effort to extract the information each time. In applications such as database systems, the type of any stored item is unlikely to change between successive runs.

The project set out to design a language where the type of a stored object is also retained. As it turns out though, it is not desirable to have a persistent data structure which is too strongly typed otherwise effort is sometimes expended unnecessarily in manipulating the objects. The project has managed to produce a firmly typed persistent programming language, PS-Algol.

Conclusion

The organisers thought to produce a questionaire so that people could give their views on the conference, especially after being asked to provide an alternative venue at the first conference. I found, and as far as I could tell the general opinion was the same, that the conference had been well organised, and had been interesting and worthwhile. Apart from Nigel Martin and his talk on NUXI, all of the talks had been aimed at imparting information, rather than delivering a sales pitch. It is interesting to find out what is actually being done within the UNIX community, rather than relying on "proprietory brand" software announcements.