THE FIRST PORT OF UNIX*

Juris Reinfelds

DEPARTMENT OF COMPUTING SCIENCE
THE UNIVERSITY OF WOLLONGONG

INTRODUCTION

In the early seventies operating systems were formidable obstacles placed between a program and its successful execution on a computer. It was firmly believed that operating systems must be large, complex and at least to some extent incomprehensible. This provided job security for an ever increasing stream of system programmers who made heroic efforts to manage the unmanagable and to comprehend the incomprehensible.

It was strongly felt (without a shred of scientific evidence one way or the other) that an operating system must be tailor made for a given computer architecture and to achieve an efficient and compact system it must be written in the assembly language of the machine.

Dijkstra had shown in 1967 that this need not be the case by designing and implementing a small, compact, comprehensible and powerful operating system called THE Operating System (for Technische Hockschule Eindhoven). His work was however either ignored or regarded as an academic curiosity especially because the system was implemented on an obscure Dutch computer EL X8.

Some universities were studying the problems of portability of operating systems but all of those systems were designed to illustrate specific problems *UNIX is a trademark of Bell Laboratories and concepts and each of them was in at least one major way insufficient as a production operating system for a computer centre service operation.

In New South Wales four separate events took place which made the first port of UNIX necessary and possible. In May 1985 I took up the Foundation Chair of Computing Science at the University of Wollongong and found that with a mainframe computer, practical laboratory work for first year students of Computing Science is neither cost effective nor flexible enough for easy use.

In July 1976 Mr. Richard Miller took up the position of Tutor in Computing Science. Richard came to us from Canada where he had taken the 20,000 line first implementation of my interactive graphics language SIGMA and produced a 2,000 line second implementation which had all the essential aspects of the first implementation but with fewer bugs and impressive running speed.

Also in 1976 the University of Wollongong allocated a small amount of money to establish a Computing Science Time Sharing Laboratory for the support of hands-on practical work by students and staff. I visited Prof. Murray Allen and Dr. John Lions at the University of N.S.W. to see how they manage their practical work and they suggested that we should take a closer look at UNIX which was available as Version 6 on PDP-11 computers.

Unfortunately (or as it later turned out fortunately) our funding was insufficient for a reasonably sized PDP-11 computer so that we purchased an Interdata (later renamed Perkin-Elmer) 7/32.

Murray Allan gave me a barely readable recopy of a copy of the first Ritchie and Thompson article on UNIX. It had appeared in the Communications of the ACM in 1974 but somehow I had missed its significance at that time. It was immediately obvious to me that here was a simple, powerful and elegant operating system which stands between the program and its execution much less than any other commercially available system that I had used or studied.

It was also obvious that the elegance of its design and the consistency of its concepts and their implications would make it possible to transfer the system to another computer where it would run efficiently provided the target machine satisfies some simple requirements such as: byte addressable memory; hardware memory mapping to provide each process with a separate address space; uniform register set.

Richard was looking for a reasonably challenging programming problem so I suggested to him that it would be terribly useful for the practical work of our students as well as a bold example for the discipline of computing science to show that elegant and simple designs are portable without loss of power, speed and capability by actually doing the port. We applied for our UNIX license and waited. Since it was a daring step to commit all resources of a fledgling Computing Science section of a Mathematics Department to a project which no one had achieved before and to expect a production level system in no more than twelve months, we decided to reduce distractions such as sceptical comments by non-believers to a minimum and talk about the project only if and when it is finished successfully.

THE FIRST STEP

Our UNIX license arrived and on November 9, 1976 Richard and I went to the University of N.S.W. to look at UNIX source code as our copy of the code had not arrived yet.

The first step was to port the C-compiler and there we had a problem. We had no PDP-11 in Wollongong and the University of N.S.W. had no Interdata 7/32.

We had the source code for the PDP-11 C-compiler which was written to generate PDP-11 assembler code. We needed a compiler which generates Interdata 7/32 assembler code. One therefore has to rewrite the code generation part of the compiler to generate assembler code for the new machine, compile this source langauge program of the compiler on the old machine (where a running compiler exists), transfer the code to the new machine test it and repeat the cycle until the compiler compiles itself on the new machine. This is a simple process if both machines are in the same room but in our case they were 80 km apart with travel funds available for at most three trips.

Richard made only two trips to UNSW and by January 5, 1977 the C-compiler was compiling itself and all the test programs we could give it without any bugs. On January 10, 1977 Ross Nealon completed the port of the UNIX editor ed to the Interdata machine.

At this stage Richard quietly put aside the optimizing pass of the C-compiler for PDP-11 code, to be implemented later when time permits. To this day nobody has complained about its absence which shows that good programs do not need automatic optimization while bad programs cannot be rescued by it.

THE KERNEL

Our Interdata 7/32 was heavily used by staff and students as a general timesharing computer seven days per week from about 8 am until 10 pm. We had no PDP-11 available to us. Hence the only way to implement the porting was to try to find a way to run the kernel of UNIX as a user process under the existing operating system of the Interdata 7/32 and to test it by simulating the Interdata Input/Output devices by software generated interrupts. Luckily this was possible in the Interdata operating system and on February 4, 1977 Richard had a working UNIX kernel without terminal drivers and interrupt handling. On February 10 we could use UNIX under the Interdata operating system with about 8 commands and a skeleton shell.

DEVICE DRIVERS

With a debugged kernel it was relatively easy to write device drivers. The main problem was inaccurate and incomplete information in device manuals which had to be rectified by experiment often with great patience and ingenuity. On April 28, 1977 at 2 pm Richard presented to us our Interdata 7/32 UNIX working in stand-alone mode without the Interdata operating system. We now had a kernel, a tty-type terminal driver, a disk driver, an interrupt handler, a shell, a few dozen system commands and ed.

PRODUCTION LEVEL SYSTEM

Migration programs were written for the conversion of Interdata files to UNIX as well as for the reverse. Some of the UNIX utilities which were written in C were ported to the Interdata and by July 1977 Interdata's FORTRAN, BASIC and CAL (assembler) were running under UNIX either with a rewritten system call interface or a system call emulator where we did not have the source code. On July 25, 1977 UNIX was put into production in the Computing Science Laboratory under a heavy user load and it was an immediate success. It is remarkable that Richard's code needed no alpha testing, no beta testing and no trial period as is customary with the introduction of a new software system. It is even more remarkable that Richard achieved stand alone operation without ever taking the machine out of production. With careful planning as much of system testing as possible was done on the simulated devices running as a user task under the Interdata system. Where standalone operation was essential Richard performed the testing late at night, between 10 pm and 8 am while carrying a full system support work load during the day. For many months night after night the light in Richard's office and in our machine room were the only bright spots in an otherwise peacefully slumbering university.

FOLLOW UP WORK

After the completion of the port other institutions became interested in our UNIX. With permission from Western Electric we shipped our UNIX to the University of Illinois where it was installed and ran for the first time in January 1978 and to the University of Melbourne in March 1978. UNIX Release 7 arrived in Wollongong on August 7, 1979 and Richard had it running in production on Saturday September 29, 1979 on our Perkin Elmer 3220 which in the meantime had replaced the Interdata 7/32.

Since we had deliberately and consciously resisted the temptation to add local ornaments, extensions and other "improvements" to the system our port of Release 7 was in production earlier than the PDP-11 versions at other Australian universities who had to convert a large amount of local modifications.

FRUSTRATIONS

In the years 1977-1980 I tried in vain to interest our industry, our university and our government in the remarkable lead that we had on the rest of the world. By 1980 we had shipped about thirty systems to all parts of the world. While on a visit to the National Computer Conference at Anaheim, California I finally met some interested people. They immediately formed a company, purchased world rights for our contribution to Perkin Elmer UNIX from the University and in honour of the occasion called the company the Wollongong Group Inc. TWG as it is now known later sold our UNIX to Perkin Elmer where it was commercially available as the first manufacturer supported UNIX called Perkin Elmer Edition 7. Only in 1984 it was superseded by Perkin Elmer's own port of UNIX System 5.2.

RETROSPECT

Bell Laboratories completed the second port in late 1977 or early 1978. The target was by coincidence another Interdata machine, the 8/32. This port made a major contribution to the enhanced portability of Release 7 UNIX but it never became a commercial product.

Richard Miller left Wollongong in 1981. In spite of his achievements the University of Wollongong was unable to provide him with a research computer dedicated to his own use so he still had to test his programs and ideas between 10 pm and 8 am....

Since leaving Wollongong, Richard has among other software developments performed several additional UNIX ports. He is probably the only person in the world who has performed five ports of four different versions of UNIX: Release 6, Release 7, Berkeley 4.1ϕ and System 5.2 to four different target machines: Interdata 7/32; Perkin Elmer 3200 series; National Semiconductor 16000 series and Motorola 68000.

UNIX porting is commonplace now but even today a port seldom exceeds in speed and elegance the six months it took Richard Miller to take a system he had never seen before to a new computer where it was not supposed to work well and to make it work better than the native operating system.

There is no better way to summarize the first port of UNIX than by quoting Dr. Doug McIlroy, the Head of the UNIX Research Group at Bell Laboratories who said

> *"We here at Bell Laboratories were truly dumfounded when this visitor from an unknown school in Australia reported his elegant procedure and remarkable success. Our own people took considerably longer to move UNIX to an Interdata machine, not because they were not as clever but because they had a different objective: a portable Unix rather than a UNIX port. But I think they'd have blinked before undertaking the heroic effort that Richard Miller did and he did not even have a Unix computer to port from."*

Thursday, June 18 9:00am – 10:30am

Joint Session: Historical UNIX

Reflections on the '73 CACM Paper
Dennis Ritchie, *Lucent Technologies, Bell Laboratories*

Dennis Ritchie will "desconstruct" the original CACM paper—what have we learned, what we got right, what did we overlook. Computing has changed in the past 25 years and the ideas of distribution and networking were less central then than they are today. Clearly, modern systems have been impacted by UNIX and the ideas presented in the original paper. But how has UNIX been impacted by our changes in thinking?

20th Anniversary of the First Port of UNIX
Steve Johnson, *Transmeta;* Richard Miller, *Miller Research;* and Juris Reinfelds, *New Mexico State University*

Nowadays, the portability of UNIX is taken for granted. The first ports of UNIX were audacious projects. Two teams independently succeeded with ports of UNIX at about the same time, only to find out about each other when the ports were finished. Both teams used different techniques for porting, and these talks will present the strategies used and how they hold up to current porting practice.

The UNIX code was so well designed that it could be picked up and ported without any consultations with the authors of the code. Reinfelds and his research team ported UNIX to Interdata 7/32 at the University of Wollongong in Australia, where Richard Miller proposed an innovative implementation of the port and proved its effectiveness by single handedly porting the kernel code and most applications. The Wollongong port later became the first computer vendor-supported UNIX.

Johnson and Ritchie were doing the port at Bell Labs on Interdata 8/32. Porting Unix required that C "get serious" about portability. Portability concerns led directly to such innovations as separate name spaces for structure members, sizeof, and indirectly to tools such as lint and social conventions such as the use of header files. This talk will also dicuss places where we were less successful, notably alignment, byte order, and bit fields.

Thursday, June 18 10:30am – 11:00am *Break*

Refereed Papers

Thursday, June 18 11:00am – 12:30pm

Performance II
Session Chair: Mike Nelson, *Silicon Graphics, Inc.*

SimICS/Sun4m: A Virtual Workstation
Peter S. Magnusson, Fredrik Larsson, Andreas Moestedt, Bengt Werner, *Swedish Institute of Computer Science;* Jim Nilsson, Par Stenstrom, Fredrik Lundholm, Magnus Karlsson, Fredrik Dahlgren, *Dept. of Computer Engineering, Chalmers Univ. of Technology;* Hakan Grahn, *Dept. of Computer Science, Univ. of Karlskrona/Ronneby*

High-Performance Caching With The Lava Hit-Server
Jochen Liedtke, Vsevolod Panteleenko, Trent Jaeger, and Nayeem Islam, *IBM*

Cheating the I/O Bottleneck: Network Storage with Trapeze/Myrinet
Darrell C. Anderson, Jeffrey S. Chase, Syam Gadde, Andrew J. Gallatin, and Kenneth G. Yocum, *Duke University;* Michael J. Feeley, *University of British Columbia*

FREENIX Track

Concurrent Session

Design and Implementation of a SCSI Subsystem
Justin T. Gibbs, *Pluto Technologies International, Inc.*

Multimedia Driver Support
James Lowe, *University of Wisconsin, Milwaukee*

Concurrent Session

ISC DHCP Distribution
Ted Lemon, *Internet Software Consortium*

Heimdal: I18N Free Kerberos Implementation
Johan Danielsson, *Royal Institute of Technology;* Assar Westerlund, *Swedish Institute of Computer Science*

Invited Talks

Software Development Models: The Cathedral and The Bazaar
Marshall Kirk McKusick, *Author and Consultant;* and Eric S. Raymond

Software projects have developed using different models to control the outcome. As software began to be developed within a world-wide culture, particularly when the source code was freely available, two distinct models of development have became popular:

- The Cathedral model, where one or two "master builders" define the basic structure of what will be built. BSD UNIX and the GNU project are typical.
- The Bazaar model is free running; there is no central control or basic building plan. Different programmers, like vendors at a bazaar, offer up different approaches to different problems. This approach has been the core of Linux development.

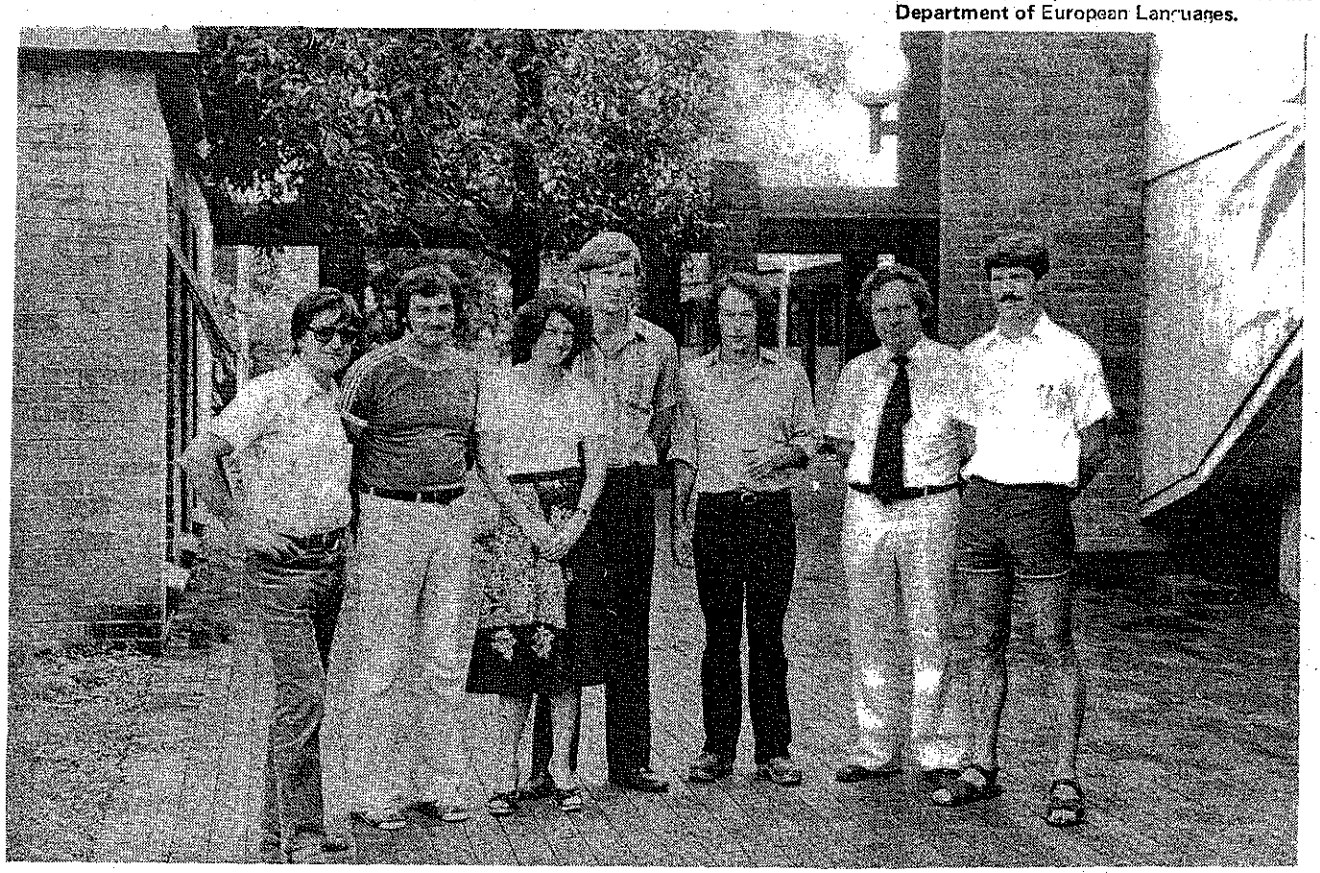
The speakers will each present their views and the history of what has worked and why. Neither Linux nor BSD was purely either approach, but the ideas presented by these two styles should make for a fascinating and lively discussion.

Thursday, June 18 12:30pm – 2:00pm *Lunch (on your own)*

NCE DEPARTMENT
EW GROUND

Campus News this year continues its description of Wollongong University departments. First on the list for 1980 is the Department of Computing Science headed by Professor J. Reinfelds.

Next issue Campus News will look at the Department of European Languages.



Members of the Computing Science Department, from left: Dr. A. Salvadori, R. Nealon, Bronwyn James (secretary), P. McKerrow, Dr. R. Dromey, Professor J. Reinfelds, Dr. T. Bailey.

Australian students."

Dr. Salvadori started out as a mathematician doing applied maths for his Master's degree and physics for his PhD. He started teaching computing science and "grew with the discipline." His particular interest lies with assimilating eco-systems ie. developing information retrieval data base systems. While in Australia he is conferring with people who work in all areas concerning the environment and examining how they organise their data. His area of research leads to more efficient ways of organising information used in all types of environmental impact studies.

* * * * * *

American-born Dr. T. Bailey has been with the department since September 1978. A graduate of Alma College, Michigan (BSc Maths and Physics) and Colorado University MSc Physics) Dr. Bailey went on to do a PhD in Computing Science at Michigan State University.

His research centres around pattern recognition ie. the different ways in which computers can be used to decide among the various possibilities it might be observing. His current work is looking at cluster analysis and whether it can be a valid tool of analysis. Dr. Bailey said cluster analysis was a fairly new and rudimentary tool of analysis, often used at present as a last resort in analysing broad area data. It can be applied throughout the sciences and social sciences to simplify certain situations. Data is divided into groups or clusters and then the interaction between the clusters described. The weak spot is that the basis for forming the clustering analysis may not necessarily be sound and research is aimed at developing ways of testing clustering results.

* * * * * *

Mr. P. McKerrow is a "home body" in the department - he graduated from Wollongong University College as a Bachelor of Engineering in 1972 and spent five years at Lysaghts working with electrical engineering machinery and doing a Master's degree part-time.

Meantime computers had come along and his Master's thesis was done on control systems based around computers. On joining the department at the beginning of 1979, Mr. McKerrow took on the task of establishing and maintaining the computing science laboratories.

His is a rapidly expanding task which includes the design and building of equipment for research being done by lecturers. A part-time PhD student, his research centres aroung logic state analysis which involves the design of a TV screen instrument to examine the internal workings of the computer. He is also doing research on micro-computers and micro-computer applications.

Continued overleaf