# The HERMES Data Production PC Farm


Wolfgang Wander [a] and Marc-André Funk [b],
on behalf of the HERMES software group

[a] *Univ. Erlangen-Nürnberg / Erwin-Rommel-Str. 1 / D-91058 Erlangen*

[b] *DESY Hamburg / Notkestr. 85 / D-22603 Hamburg*



Commodity computing hardware nowadays offers large gains in price/performance ratio and therefore becomes more and more attractive to HEP experiments. A large increase in the computing power requirements for the HERMES experiment at DESY forced us to double our available offline CPU resources. Data production so far had been done on a 28-Processor SGI Challenge. Investigation of the price/performance ratio of various upgrade solutions resulted in the decision to build up a PC farm of 10 PPro systems in Sep 96. The choice of the operating system was based on the requirements for data and Monte-Carlo production and led to the Unix flavour Linux. Here we showed that unlike on other PC operating systems (like Windows NT) the overhead of porting the available software base to a new operating system is small to neglectable. Two men-weeks of porting the 600000 lines of HERMES software resulted in a first production run of the farm already one week after the hardware installation. The performance of the PPro processor together with the better system performance of the Linux operating system decreases the reconstruction time per CPU by more than a factor of 2 over a single CPU in the SGI Challenge. During Jan/Feb 97 the PC Farm has run the first pass of the HERMES 96 data production within the expected performance.



*Key words:* Commodity hardware; PC farm; Linux-Cluster; High-Performance Computing;


## 1 Motivation

In 1996 the HERMES experiment was faced with a large increase of CPU requirements. HERMES is a high statistics high energy experiment at the HERA-collider at DESY Hamburg, investigating the internal spin structure of

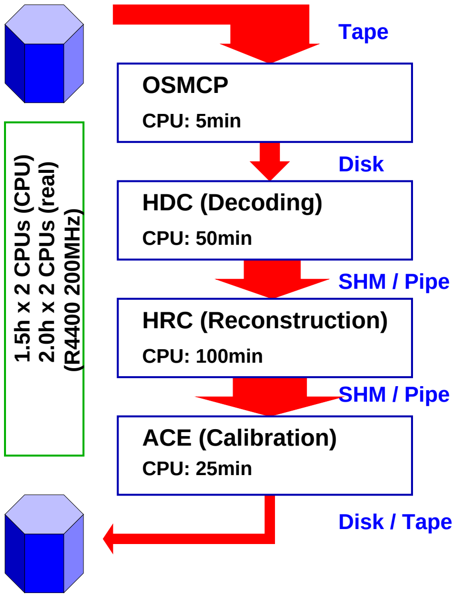


Fig. 1. Modular Layout of the HERMES data production. Raw EPIO data files are copied from the Mass-Storage system to disk where the decoder calibrates and converts this data into ADAMO[3] format and pipes it to the reconstruction and a on-the-fly calibration process. The output of the production is stored on disks and tapes.

protons and neutrons, by scattering a polarised electron beam off a polarised internal gas target.

The data volume recorded in 1996 corresponds to 4TB of raw data or $400 \cdot 10^6$ events stored in 8000 run files on the DESY Mass-Storage systems[1]. During data production DSTs containing reconstructed track information are generated from the raw data and are stored on large DST disk arrays. The modular layout of the production is shown in figure 1. A highly optimised reconstruction technique[2] allows for a quick data production process and several iterations of it to optimise calibration constants of the detector. In the past this data production was hosted on a 28-cpu (R4400 150/200MHz) SGI Challenge. At nearly 100% system usage this process took about 8–10 weeks. As this machine was the only analysis machine at DESY for the collaboration, HERMES planned to double its CPU resources, moving the data production to a new machine and utilising the SGI for interactive user analysis.

## 2 System Alternatives

HERMES supported up to now three different operating systems: DEC Unix, Ultrix and Irix. Avoiding software porting costs and doubling the available CPU resources with DEC Unix (Alpha based) or IRIX (SGI based) would have

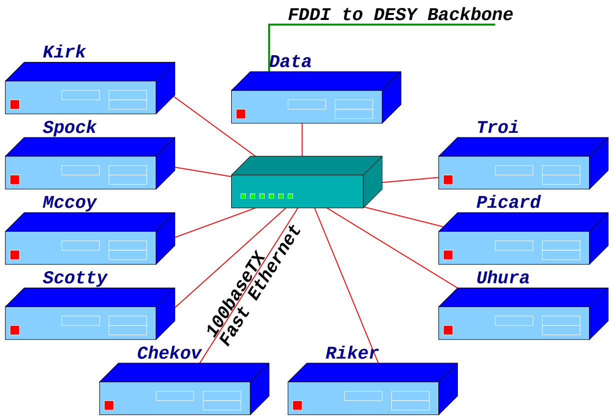


Fig. 2. The layout of the PC farm. The cluster is connected via a Fast-Ethernet segment. One PC serves as an Ethernet-FDDI router to the DESY backbone.

involved an investment of 250 – 600kDM (autumn 96). On the other hand a PC based production farm with the same CPU performance but another operating system was available for 70kDM at that time. Thus the effort of porting the HERMES software seemed to be well acceptable.

The PC farm configuration chosen was a cluster of 10 PPro/200MHz Systems each equipped with two 4GB SCSI disks and 128 MByte memory. The cluster resides on a Fast-Ethernet Segment, one PC serves as a router for an FDDI connection to the DESY network backbone.

## 3 The Operating System

There exists a huge variety of operating systems for PCs. The most common ones are Windows (3.1/95), Windows NT, OS/2 and various Unix dialects: Solaris, SCO Unix, Free-BSD, Linux... The decision for a PC operating system becomes more easy taking the availability of the CERN libraries in account: Only Windows NT and Linux ports are available.

HERMES chose Linux as the operating system for the farm for various reasons:

- Least important: No software licences have to be bought for Linux, although there are powerful commercial tools available. The licences for NT amount to about 500–1000DM/machine, including the operating system, compilers, network licenses.
- The development environments on NT are excellent. However they are not compatible to the so far used UNIX tools. At the time we started there was no operational debugger for Fortran code available which was able to handle the large executables. On the other hand Linux comes with most standard Unix tools, with somewhat reduced functionality for Fortran77.

- Porting the HERMES software to NT was assumed to require 0.5 to 1 men-years, based on the experience of other HEP experiments. Porting the software to Linux was expected to take and took about 2 men-weeks.
- The CPU performance of both systems is equivalent. However tests have shown a poor Fast-Ethernet performance for NT which also put a high CPU load on the system (2MB/s at 90% system load). Linux performs at 10MB/s at 10% system load.
- The above mentioned reconstruction chain needs an optimal multitasking to distribute the CPU resources to the three different processes. Here NT's multitasking did not seem to be sufficient.

## 4 The Linux Operating System

Linux is a Unix clone written by Linus Torvalds and developers across the Internet. It aims towards POSIX compliance. In general Linux provides a very stable and highly optimised kernel. It has all features of a modern Unix Operating System: true multitasking, virtual memory, shared libraries, demand loading, shared copy-on-write executables, proper memory management, SMP and TCP/IP networking. It mainly runs on x86-based PCs. Various other architectures are also supported or being worked on: DEC Alpha, MIPS, Sparc, PPC, 680x0.

Linux itself is copyrighted under the GPL (GNU Public Licence) and also profits a lot from GNU developments. Without the large number of GNU tools and programs, like e.g. gcc, Linux would not have been possible.

Various free utilities are available, among them compilers, editors, networking, TeX, X-Windows. An emulator for DOS programs runs on Linux as well as an emulation for SCO binaries. There is a growing commercial interest of the Unix software market in Linux as it has become the Unix dialect with the largest number of installations world-wide. Linux runs as AFS client, WWW Server, Firewall in Netware environments. Commercial applications like WordPerfect, Staroffice, Sniff++, Maple, Mathematica ... are available and make this operating system an attractive platform also for the scientific user.

The CERN libraries have been ported as well as ROOT, a possible successor for the CERN libraries. Experiments and institutes like NA43, NA49, CMS, BES, Crystal Barrel, Hera-B, BaBar, LSND, QMW London, Niels Bohr Institute Copenhagen use Linux for scientific applications in High-Energy Physics.

## 5 Porting the HERMES Software

HERMES software consist of 600000 lines of code: 50% C code and 40 % Fortran code, the rest are shell and PinK (tcl/tk) scripts. Whereas there were no problems to compile the ANSI C code with gcc and also no problems to get shell and tcl/tk scripts, porting the HERMES Fortran code was a little more problematic.

The free Linux Fortran compilers `g77` and `f2c` require a strict Fortran77 code. Though the HERMES coding rules require strict Fortran77 compliance, the so far non-existing tests for this standard resulted in several violations.

Common Fortran extensions, like omitted requirements for the order of data statements and the initialisation of variables in their declaration, are automaticly corrected by a program named `f77reorder`, developed at HERMES and used as an additional preprocessor for the code. Only a few exceptions thus had to be handled manually. Debugging Fortran code is also not completely supported by `g77` and `f2c`: there is no way to access variables within common blocks (commercial compilers like pgf77 compiler (Portland Group) do not suffer from this problem).

As a result of the porting efforts the HERMES software tree got completely translated to Linux after 2 men-weeks, offering a new supported platform to the collaboration members with excellent price/performance features.

## 6 Data Production on the PC Farm

Data Production on the farm has to handle the job distribution on the farm and disk allocation on the DST disk arrays for the production output. This requirement is very similar to the ones of a standard slow-control task, therefore the HERMES slow-control scheme[4–6] was used also for the job control.

A controller job named `spooler` communicates with `daemons` running on each of the PCs. It demands the loading of raw data files from the mass-storage robots to the staging disks of the PC and the execution of the jobs. The loading of raw data and the processing of run files is handled in parallel to avoid the long delay times for the tape access. The communication between `spooler` and `daemon` is illustrated in figure 3.

Monitoring jobs, written with the standard HERMES tool PiNK[5] are available which control the status and performance of the cluster (see figure 4).

In December 1996 the second iteration started on the PC farm. After initial

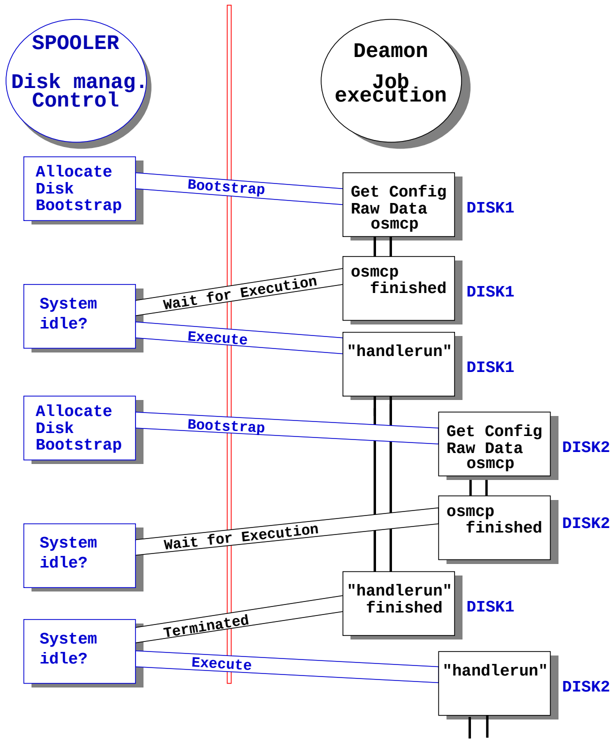


Fig. 3. Communication between controller and one of the PCs. In the beginning the PC is asked to load the configuration files and the raw data file to its disk. After this is done, the job is executed and in parallel a new run file is copied to the stage disk. The execution of the new runs is launched after the first one has terminated.

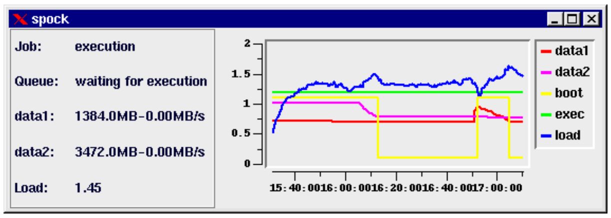


Fig. 4. Monitoring of the cluster components is done with PiNK clients. They provide information about the system status and performance over a given period of time.

problems with the FDDI-Ethernet switch one of the PCs was configured to do the routing. Since then the farm ran at 98% cpu usage for 7 weeks without any interruption.

## 7 Conclusion

The HERMES PC farm provides the expected excellent performance, the production was moved from the SGI multiprocessor machine to an inexpensive farm of commodity PCs. The porting of the existing HERMES software to

Linux did not require more resources than to another standard Unix system.

In addition this step offers a new inexpensive platform to the HERMES collaborators. Various PCs have already been installed at outside institutes or are being ordered.

The near future of the PC farm will be an upgrade of the PCs to Dual PPro Machines using Linux' SMP features. This will nearly double the available CPU resources for another 20kDM.

## 8 Acknowledgements

A large contribution to the success of this project came from the DESY central computing division. We had a qualified and quick support for all operating, network and mass-storage related questions.

We also have to thank Linus Torvalds and the whole Linux developers community for the work on the operating system and the GNU project for the GNU utilities and compilers. This must also include AT&T for their `f2c` project.

This work was supported in part by BMBF, Germany under contract 056ER12I(3) and 057ER12P(2).

## References

[1] P. Fuhrmann: **A Perfectly Normal Namespace for the DESY Open Storage Manager,** Submitted to CHEP 97, Berlin, 7.4.–11.4.1997.

[2] W. Wander: **Rekonstruktion hochenergetischer Streuereignisse im HERMES Experiment,** PhD Thesis, Universität Erlangen-Nürnberg (1996).

[3] Programming Techniques Group: **ADAMO – Entity-Relationship Programming System, Version 3.3,** ECP Division, CERN, Geneva (1994).

[4] W. Wander et al.: **DAD – Distributed ADAMO Database system at Hermes,** Proc. of CHEP 95, Rio de Janeiro, 18.9.–22.9.1995, 349.

[5] K. Ackerstaff, M.A. Funk et al.: **PiNK: A TCL/Tk based Database Interface to ADAMO and DAD,** Proc. of CHEP 95, Rio de Janeiro, 18.9.–22.9.1995, 354.

[6] R. Kaiser et al.: **Online Monitoring at HERMES,** Submitted to CHEP 97, Berlin, 7.4.–11.4.1997.