Building Beowulfs for High Performance Computing

Duncan A. Grove, K. A. Hawick, P. D. Coddington and F. A. Vaughan
Advanced Computational Systems Cooperative Research Centre
Department of Computer Science, University of Adelaide
Adelaide, SA 5005, Australia
{*duncan,khawick,paulc,francis*}*@cs.adelaide.edu.au*

19 June 1999

Abstract

Access to High Performance Computing (HPC) power remains crucial in many scientific, engineering and business applications. In some sectors demand is growing. Unfortunately, traditional supercomputers or cluster computing solutions from commercial vendors are very expensive. They are beyond the reach of many small or even medium sized organisations, universities and research groups. A Beowulf class computer cluster can provide an affordable alternative with many advantages.

This paper introduces the concept of Beowulf class computers and provides a brief justification for using Linux in such systems. It focuses on actually building a working Beowulf class computer, from initial design choices and an early prototype right through to getting real results from production codes. The basis for this discussion is a case study of a large Linux-powered Beowulf being built for computational chemistry at the University of Adelaide. Details of the cost and performance advantages for a Linux based system on presently available platforms are discussed.

Keywords: Beowulf, Linux, HPC, supercomputer, cluster computing.

1 Introduction

- The Beowulf concept
- The fundamental paradigms for parallel computing
- A brief history of cluster computing

2 Why Linux?

- An objective comparison with other systems

- The part Linux & free source is playing in developing Beowulf
- It's good: performance, availability, flexibility
- Large, active user community

3 Which Hardware?

- There are many choices, both economic and technical
- Which platform?
- Which interconnect?
- Storage: RAM and disk?

4 Setting Things Up

- Getting some machines
- Into the depths of network configuration
- Streamlining the base installation
- Managing a collection of machines
- Installing further software
- Maintaining the system

5 Which Software?

- Compilers and development tools, message passing APIs
- Queueing software
- User Environment
- Other parallel and distributed software technologies

6 Performance

- Batch job throughput
- Parallel program performance

7 Conclusions

- Beowulfs make high performance computing more affordable
- Growing software base at both the application and system level
- Linux is enabling this
- Linuxwulves will be everywhere!

Acknowledgements

Thanks to Cheryl and Michael Pope for helping us rescue Beowulf nodes from the scrap heap and to Mark Buntine and his consortium of computational chemists at Adelaide University, Flinders University and the University of South Australia for whom we are constructing a large Beowulf system. This work was carried out under the Distributed High-Performance Computing Infrastructure (DHPC-I) project of the Research Data Networks (RDN) and Advanced Computational Systems (ACSys) Cooperative Research Centres (CRC). RDN and ACSys are established under the Australian Government's CRC Program.

References

[1] Thomas Stirling, Donald J. Becker and Daniel Savarese, *BEOWULF: A PARALLEL WORKSTATION FOR SCIENTIFIC COMPUTATION.*

[2] DHPC Beowulf Project Web pages: *http://dhpc.adelaide.edu.au/projects/beowulf*