

How to Build a Beowulf

A Guide to the Implementation and Application of PC Clusters

Thomas L. Sterling

John Salmon

Donald J. Becker

Daniel F. Sussman

The MIT Press
Cambridge, Massachusetts
London, England

Contents

Series Foreword	xiii
Foreword	xv
Preface	xvii
1 Introduction	1
1.1 A Brief History	2
1.2 The Bowell Basis	4
2 Overview of Bowell Systems	9
2.1 What Is a Bowell?	9
2.2 A Taxonomy of Parallel Computing	12
2.3 Benefits of Bowell	13
2.4 A Critical Technology Convergence	14
2.5 The Bowell System Node	16
2.6 The Bowell Network	16
2.7 Linux	19
2.8 Message Passing for Interprocessor Communication	20
2.9 Bowell System Management	21
2.10 The Bowell Challenge	21
3 Node Hardware	25
3.1 Overview of a Bowell Node	26
3.1.1 Principal Specifications	28
3.1.2 Basic Elements	29
3.2 Firmware	32
3.2.1 Intel Pentium II	33
3.2.2 DEC Alpha	34
3.2.3 AMD 586	34
3.2.4 Future Firmware	34

3.3	Motherboard	26
3.3.1	A Typical Internal Motherboard	27
3.3.2	Selection Considerations	37
3.3.3	LAYOUT of Major Components	38
3.4	Memory	43
3.4.1	Memory Capacity	43
3.4.2	Memory Speed	42
3.4.3	Memory Types	43
3.4.4	Memory Hierarchy—L1 and L2 Caches	44
3.4.5	Parallels Bytes	46
3.5	BIOS	46
3.6	Secondary Storage	47
3.7	PCI Bus	48
3.8	Examples of a General Node	49
3.8.1	Intel Pentium II Based Node	50
3.8.2	DEC Alpha Based Node	50
3.9	Buses, Slots, Pins, and Backs	51
3.10	Node Assembly	52
3.10.1	Motherboard Preparation	54
3.10.2	The Case	55
3.10.3	Internal Peripherals	56
3.10.4	Powering the System	57
3.10.5	Installing the Other Components	59
3.10.6	Transferring a PC that Won't Boot	60
4	The Linux Operating System	63
4.1	History of Linux	64
4.2	Linux Kernels and Linux Distributions	66
4.3	Linux Features	67
4.4	File Systems	70
4.5	System Configuration	73
4.6	Tools for Program Development	76

4.7	Linux's Unique Features	83
4.8	Installing an Initial System	84
4.8.1	Installation by CD or Network?	85
4.8.2	Installation Summary	86
4.9	Keeping Up with Linux	88
4.10	Suggested References	91
4.10.1	Web Sites	91
4.10.2	Books	92
5	Network Hardware and Software	93
5.1	Ethernet	93
5.1.1	Ethernet Architecture	94
5.1.2	Packet Format	95
5.1.3	NIC Architecture	97
5.1.4	Physical Switches	99
5.1.5	Gigabit Ethernet	101
5.2	Alternative Network Technologies	102
5.2.1	Asynchronous Transfer Mode	102
5.2.2	FSM	103
5.2.3	Scalable Coherent Interface	104
5.2.4	Myrinet	104
5.3	TCP/IP	105
5.3.1	IP Addressing	106
5.3.2	Zero-copy Protocols	107
5.4	Networks	108
5.5	Higher Level Protocols	112
5.5.1	Remote Procedure Calls	113
5.5.2	Distributed Objects: CORBA and Java RMI	115
5.6	Distributed File Systems	118
5.6.1	NFS	118
5.6.2	AFS	119
5.6.3	Autofs: The Automounter	120

5.7	Remote Command Execution	120
5.7.1	RSD-R Commands	120
5.7.2	RSH—The Secure Shell	121
6	Managing Ensembles	123
6.1	System Access Models	123
6.1.1	The Stand-alone System	123
6.1.2	The Universally Accessible Machine	124
6.1.3	The Guarded Resource	124
6.2	Assigning Names	125
6.2.1	Dynamically Assigned Addresses	126
6.3	Cloning Nodes	127
6.3.1	Cloning The Images	128
6.3.2	Setting up a Clone Host Partition	128
6.3.3	Setting up BOOTP	129
6.3.4	Building a Boot Clone Floppy	131
6.4	Basic System Administration	132
6.4.1	Booting and Shutting Down	133
6.4.2	The Node File System	133
6.4.3	Account Management	135
6.4.4	PSH—Parallel Remote Shell	136
6.5	Defending the Pack: Security Strategies	137
6.5.1	System Configuration	138
6.5.2	IP Masquerading	138
6.5.3	Restricting Host Access	141
6.5.4	RSH: Secure Shell Revisited	142
6.6	Job Scheduling	143
7	Parallel Applications	145
7.1	Parallelism	145
7.2	Broad Categories of Parallel Algorithms	148
7.2.1	Regular and Irregular	148

7.2.2	Synchronous and Asynchronous	149
7.2.3	Coarse and Fine Grained	149
7.2.4	Rowlock's Greedy and Pregel	150
7.2.5	Latency Tolerant and Indifferent	151
7.2.6	Distributed and Hetero Address Spaces	152
7.2.7	Resilient Systems and Choice of Parallelism	153
7.3	Process-level Parallelism	153
7.3.1	Example: Ray Tracing Animation	154
7.3.2	Utilities for Process-parallel Computing	156
7.3.3	Overheads – SSH and File I/O	159
7.3.4	Summary	160
8	MPI – A User-level Message-passing Interface	163
8.1	History	163
8.2	MPI Basic Functionality	164
8.2.1	Example: “Hello World” in MPI	165
8.3	Parallel Data Structures with MPI	173
8.3.1	Example: A Parallel Array	174
8.3.2	Example: A One-dimensional Cellular Automaton	180
8.4	MPI Advanced Features	181
8.4.1	Blocking and Non-blocking Calls and Alternative Sending Modes	185
8.4.2	Virtual Topologies and Attribute Caching	187
8.4.3	Derived Data Types	187
8.4.4	Intercommunicators	188
8.4.5	MPI-2	188
9	Programming with MPI – A Detailed Example	191
9.1	Example: Sorting a List of Uniformly Distributed Integers	193
9.2	Analysis of Integer Sort	198
9.2.1	Communication	199
9.2.2	Rehearsal	200
9.2.3	Extra Work	200

9.2.4	Load Imbalance	266
9.2.5	Waiting	269
9.3	Measurement of Integer Sort	269
9.4	Example: Sorting with User-supplied Comparator	269
9.5	Analysis of a More General Sort	272
9.5.1	Choosing the Number of Recursions	273
9.6	Summary	274
10	Conclusions and Views	277
10.1	New Generation Demands	277
10.2	New Opportunities	277
10.2.1	Databases	277
10.2.2	Web Servers	277
10.2.3	Dynamic Interactive Simulation	277
10.2.4	Virtual Reality	277
10.2.5	Process Control	277
10.2.6	Artificial Intelligence	277
10.2.7	Genetic Programming	277
10.3	Time Costs	277
10.4	Total Work versus Peak Performance	277
10.5	Big Memory versus Optimization	277
10.6	Outstanding Influences on the Overall Demands	277
10.7	New Programming Models	277
10.8	Will Linux Survive the Next Market?	277
10.9	Final Thoughts	277