

Contents

Series Foreword	xiii
Preface to the Second Edition	xv
Preface to the First Edition	xix
1 Background	1
1.1 Why Parallel Computing?	1
1.2 Obstacles to Progress	2
1.3 Why Message Passing?	3
1.3.1 Parallel Computational Models	3
1.3.2 Advantages of the Message-Passing Model	9
1.4 Evolution of Message-Passing Systems	10
1.5 The MPI Forum	11
2 Introduction to MPI	13
2.1 Goal	13
2.2 What Is MPI?	13
2.3 Basic MPI Concepts	14
2.4 Other Interesting Features of MPI	18
2.5 Is MPI Large or Small?	20
2.6 Decisions Left to the Implementor	21
3 Using MPI in Simple Programs	23
3.1 A First MPI Program	23
3.2 Running Your First MPI Program	28
3.3 A First MPI Program in C	29
3.4 A First MPI Program in C++	29
3.5 Timing MPI Programs	34
3.6 A Self-Scheduling Example: Matrix-Vector Multiplication	35
3.7 Studying Parallel Performance	43
3.7.1 Elementary Scalability Calculations	43
3.7.2 Gathering Data on Program Execution	45
3.7.3 Instrumenting a Parallel Program with MPE Logging	46

3.7.4	Events and States	47
3.7.5	Instrumenting the Matrix-Matrix Multiply Program	47
3.7.6	Notes on Implementation of Logging	49
3.7.7	Examining Logfiles with Upshot	52
3.8	Using Communicators	53
3.9	Another Way of Forming New Communicators	59
3.10	A Handy Graphics Library for Parallel Programs	62
3.11	Common Errors and Misunderstandings	64
3.12	Application: Quantum Monte Carlo Calculations in Nuclear Physics	66
3.13	Summary of a Simple Subset of MPI	67
4	Intermediate MPI	69
4.1	The Poisson Problem	70
4.2	Topologies	73
4.3	A Code for the Poisson Problem	81
4.4	Using Nonblocking Communications	93
4.5	Synchronous Sends and “Safe” Programs	96
4.6	More on Scalability	96
4.7	Jacobi with a 2-D Decomposition	98
4.8	An MPI Derived Datatype	100
4.9	Overlapping Communication and Computation	101
4.10	More on Timing Programs	105
4.11	Three Dimensions	107
4.12	Common Errors and Misunderstandings	107
4.13	Application: Simulating Vortex Evolution in Superconducting Materials	109
5	Advanced Message Passing in MPI	111
5.1	MPI Datatypes	111
5.1.1	Basic Datatypes and Concepts	111
5.1.2	Derived Datatypes	114

5.1.3	Understanding Extents	117
5.2	The N-Body Problem	117
5.2.1	Gather	118
5.2.2	Nonblocking Pipeline	123
5.2.3	Moving Particles between Processes	124
5.2.4	Sending Dynamically Allocated Data	132
5.2.5	User-Controlled Data Packing	134
5.3	Visualizing the Mandelbrot Set	138
5.3.1	Sending Arrays of Structures	145
5.4	Gaps in Datatypes	146
5.4.1	MPI-2 Functions for Manipulating Extents	148
5.5	New MPI-2 Datatype Routines	150
5.6	More on Datatypes for Structures	152
5.7	Deprecated Functions	154
5.8	Common Errors and Misunderstandings	156
6	Parallel Libraries	157
6.1	Motivation	157
6.1.1	The Need for Parallel Libraries	157
6.1.2	Common Deficiencies of Previous Message-Passing Systems	158
6.1.3	Review of MPI Features That Support Libraries	160
6.2	A First MPI Library	163
6.2.1	MPI-2 Attribute-Caching Routines	172
6.2.2	A C++ Alternative to MPI_Comm_dup	172
6.3	Linear Algebra on Grids	177
6.3.1	Mappings and Logical Grids	178
6.3.2	Vectors and Matrices	181
6.3.3	Components of a Parallel Library	185
6.4	The LINPACK Benchmark in MPI	189
6.5	Strategies for Library Building	190
6.6	Examples of Libraries	192

7	Other Features of MPI	195
7.1	Simulating Shared-Memory Operations	195
7.1.1	Shared vs. Distributed Memory	195
7.1.2	A Counter Example	196
7.1.3	The Shared Counter Using Polling instead of an Extra Process	200
7.1.4	Fairness in Message Passing	201
7.1.5	Exploiting Request-Response Message Patterns	202
7.2	Application: Full-Configuration Interaction	205
7.3	Advanced Collective Operations	206
7.3.1	Data Movement	206
7.3.2	Collective Computation	206
7.3.3	Common Errors and Misunderstandings	213
7.4	Intercommunicators	214
7.5	Heterogeneous Computing	220
7.6	The MPI Profiling Interface	222
7.6.1	Finding Buffering Problems	226
7.6.2	Finding Load Imbalances	228
7.6.3	The Mechanics of Using the Profiling Interface	228
7.7	Error Handling	229
7.7.1	Error Handlers	230
7.7.2	An Example of Error Handling	233
7.7.3	User-Defined Error Handlers	234
7.7.4	Terminating MPI Programs	237
7.7.5	MPI-2 Functions for Error Handling	239
7.8	The MPI Environment	240
7.8.1	Processor Name	242
7.8.2	Is MPI Initialized?	242
7.9	Determining the Version of MPI	243
7.10	Other Functions in MPI	245
7.11	Application: Computational Fluid Dynamics	246
7.11.1	Parallel Formulation	246

7.11.2	Parallel Implementation	248
8	Understanding how MPI Implementations Work	253
8.1	Introduction	253
8.1.1	Sending Data	253
8.1.2	Receiving Data	254
8.1.3	Rendezvous Protocol	254
8.1.4	Matching Protocols to MPI's Send Modes	255
8.1.5	Performance Implications	256
8.1.6	Alternative MPI Implementation Strategies	257
8.1.7	Tuning MPI Implementations	257
8.2	How Difficult Is MPI to Implement?	257
8.3	Device Capabilities and the MPI Library Definition	258
8.4	Reliability of Data Transfer	259
9	Comparing MPI with Other Systems for Interprocess Communication	261
9.1	Sockets	261
9.1.1	Process Startup and Shutdown	263
9.1.2	Handling Faults	265
9.2	PVM 3	266
9.2.1	The Basics	267
9.2.2	Miscellaneous Functions	268
9.2.3	Collective Operations	268
9.2.4	MPI Counterparts of Other Features	269
9.2.5	Features Not in MPI	270
9.2.6	Process Startup	270
9.2.7	MPI and PVM related tools	271
9.3	Where to Learn More	272
10	Beyond Message Passing	273
10.1	Dynamic Process Management	274
10.2	Threads	275

10.3	Action at a Distance	276
10.4	Parallel I/O	277
10.5	MPI-2	277
10.6	Will There Be an MPI-3?	278
10.7	Final Words	278
	Glossary of Selected Terms	279
A	Summary of MPI-1 Routines and Their Arguments	289
B	The MPICH Implementation of MPI	329
C	The MPE Multiprocessing Environment	337
D	MPI Resources on the World Wide Web	345
E	Language Details	347
	References	353
	Subject Index	363
	Function and Term Index	367