
MPI - Eine Einführung

Portable parallele Programmierung mit dem
Message-Passing Interface

von

William Gropp, Ewing Lusk, Anthony Skjellum

Übersetzt von Dr. Holger Blaar, Halle

Wissenschaftliche Leitung der Übersetzung:
Prof. Dr. Paul Molitor,
Martin-Luther-Universität Halle-Wittenberg

Oldenbourg Verlag München Wien

Inhaltsverzeichnis

1	Motivation	1
1.1	Parallelverarbeitung – warum?	1
1.2	Hemmnisse	2
1.3	Warum Message-Passing?	3
1.3.1	Parallele Berechnungsmodelle	3
1.3.2	Vorteile des Message-Passing-Modells	8
1.4	Entwicklung der Message-Passing-Systeme	9
1.5	Das MPI-Forum	10
2	Einführung in MPI	13
2.1	Ziel	13
2.2	Was ist MPI?	13
2.3	Grundlegende MPI-Konzepte	14
2.4	Weitere interessante Konzepte in MPI	18
2.5	Ist MPI umfangreich oder knapp?	20
2.6	Dem Entwickler verbleibende Entscheidungen	21
3	MPI in einfachen Programmen	23
3.1	Ein erstes MPI-Programm	23
3.2	Ausführung des ersten MPI-Programms	28
3.3	Ein erstes MPI-Programm in C	29
3.4	Ein erstes MPI-Programm in C++	29
3.5	Zeitmessung in MPI-Programmen	33
3.6	Ein Beispiel für autonome Prozesskoordination	35
3.7	Untersuchung der parallelen Leistung	42
3.7.1	Elementare Berechnungen zur Skalierbarkeit	42
3.7.2	Aufzeichnung von Informationen über die Programmausführung	44
3.7.3	MPE-Protokollierung in parallelen Programmen	45
3.7.4	Ereignisse und Zustände	46

3.7.5	Protokollierung im Programm zur Matrizenmultiplikation	46
3.7.6	Bemerkungen zur Implementierung der Protokollierung	48
3.7.7	Aufbereitung der Logdateien mit Upshot.....	51
3.8	Die Arbeit mit Kommunikatoren.....	52
3.9	Ein anderer Weg zur Bildung neuer Kommunikatoren	60
3.10	Eine praktische Graphikbibliothek für parallele Programme	61
3.11	Typische Fehler und Missverständnisse	63
3.12	Anwendung: Quanten-Monte-Carlo Berechnungen in der Kernphysik ...	65
3.13	Zusammenfassung	66
4	MPI für Fortgeschrittene	67
4.1	Das Poisson-Problem	68
4.2	Topologien	71
4.3	Ein Programm für das Poisson-Problem.....	80
4.4	Anwendung nichtblockierender Kommunikationen	89
4.5	Synchrones Senden und „sichere“ Programme	91
4.6	Mehr zur Skalierbarkeit	92
4.7	Jacobi-Verfahren mit 2D-Dekomposition	95
4.8	Ein erweiterter MPI-Datentyp	96
4.9	Überlagern von Kommunikation und Berechnung.....	99
4.10	Mehr zur Laufzeitmessung in Programmen	101
4.11	Drei Dimensionen	103
4.12	Typische Fehler und Missverständnisse	104
4.13	Simulation der Wirbelevolution in supraleitenden Materialien	105
5	Weitergehende Details zum Message-Passing mit MPI	107
5.1	MPI-Datentypen	107
5.1.1	Basisdatentypen und Konzepte	107
5.1.2	Abgeleitete Datentypen	112
5.1.3	Was genau ist die Länge?	113
5.2	Das N-Körper-Problem	113
5.2.1	Aufsammeln von Informationen	115
5.2.2	Nichtblockierende Pipeline.....	118
5.2.3	Verschieben von Teilchen zwischen Prozessen	121
5.2.4	Senden dynamisch erzeugter Daten	127

5.2.5	Nutzergesteuertes Packen von Daten	129
5.3	Visualisierung der Mandelbrotmenge	131
5.3.1	Senden von Feldern von Strukturen	139
5.4	Lücken in Datentypen	140
5.4.1	Funktionen in MPI-2 zur Manipulation der Länge eines Datentyps	141
5.5	Neue Datentyp-Routinen in MPI-2	143
5.6	Weiteres zu Datentypen für Strukturen	145
5.7	Veraltete Funktionen	148
5.8	Typische Fehler und Missverständnisse	149
6	Parallele Bibliotheken	151
6.1	Motivation	151
6.1.1	Die Notwendigkeit paralleler Bibliotheken	151
6.1.2	Bekannte Schwächen älterer Message-Passing-Systeme	152
6.1.3	Überblick zu MPI-Eigenschaften für die Bibliotheksunterstützung	154
6.2	Eine erste MPI-Bibliothek	157
6.2.1	Routinen für das Attribut-Caching in MPI-2	165
6.2.2	Eine Alternative in C++ zu MPI_Comm_dup	165
6.3	Lineare Algebra auf Gittern	170
6.3.1	Abbildungen und Logische Gitter	171
6.3.2	Vektoren und Matrizen	176
6.3.3	Komponenten einer parallelen Bibliothek	178
6.4	Der LINPACK-Benchmark mit MPI	181
6.5	Strategien für die Erstellung von Bibliotheken	182
6.6	Beispiele für Bibliotheken	183
7	Weitere Eigenschaften von MPI	185
7.1	Simulation von Shared-Memory-Operationen	185
7.1.1	Gemeinsamer und verteilter Speicher	185
7.1.2	Ein Zähler-Beispiel	186
7.1.3	Realisierung des gemeinsamen Zählers per Polling anstelle eines gesonderten Prozesses	190
7.1.4	Fairness im Message-Passing	191
7.1.5	Ausnutzung von Anfrage-Antwort-Mustern	192
7.2	Die vollständige Konfigurationswechselwirkung als Anwendungsbeispiel ..	195
7.3	Erweiterte kollektive Operationen	196
7.3.1	Bewegen von Daten	196
7.3.2	Kollektive Berechnung	196
7.3.3	Typische Fehler und Missverständnisse	202

7.4	Interkommunikatoren	203
7.5	Rechnen in einer heterogenen Umgebung	210
7.6	Die MPI-Schnittstelle zur Programmanalyse	210
7.6.1	Entdecken von Pufferproblemen	214
7.6.2	Erkennung ungleichmäßiger Lastverteilung	216
7.6.3	Der Mechanismus zur Nutzung der Analyseschnittstelle	217
7.7	Fehlerbehandlung	218
7.7.1	Routinen zur Fehlerbehandlung	218
7.7.2	Ein Beispiel zur Fehlerbehandlung	221
7.7.3	Anwenderdefinierte Fehlerbehandlungen	223
7.7.4	Abbruch von MPI-Programmen	224
7.7.5	Funktionen in MPI-2 zur Fehlerbehandlung	226
7.8	Die MPI-Umgebung	227
7.8.1	Prozessorname	229
7.8.2	Ist MPI initialisiert?	229
7.9	Ermittlung der Version von MPI	230
7.10	Weitere Funktionen in MPI	231
7.11	Eine Anwendung: Numerische Strömungsmechanik	232
7.11.1	Parallelisierung	233
7.11.2	Parallele Implementierung	235
8	Wie arbeiten MPI-Implementierungen?	239
8.1	Einführung	239
8.1.1	Senden von Daten	240
8.1.2	Empfangen von Daten	240
8.1.3	Rendezvous-Protokoll	241
8.1.4	Zuordnung zwischen Protokollen und MPI-Sendemodi	241
8.1.5	Auswirkungen auf die Leistung	242
8.1.6	Alternative Strategien für MPI-Implementierungen	243
8.1.7	Anpassung von MPI-Implementierungen	243
8.2	Wie schwierig ist es, MPI zu implementieren?	244
8.3	Zusammenspiel von Hardware-Eigenschaften und MPI-Bibliothek	244
8.4	Sicherheit der Datenübertragung	245
9	MPI im Vergleich	247
9.1	Sockets	247
9.1.1	Erzeugen und Beenden von Prozessen	249
9.1.2	Behandlung von Fehlern	251
9.2	PVM 3	252
9.2.1	Grundlagen	253

9.2.2	Weitere Funktionen	254
9.2.3	Kollektive Operationen	254
9.2.4	MPI-Gegenparts weiterer PVM-Funktionalitäten	255
9.2.5	Funktionalitäten, die sich in MPI nicht wiederfinden	256
9.2.6	Starten von Prozessen	256
9.2.7	Zu MPI und PVM ähnliche Werkzeuge	257
9.3	Wo kann man noch mehr erfahren?	257
10	Über Message-Passing hinaus	259
10.1	Dynamische Prozessverwaltung	260
10.2	Threads	261
10.3	Ausführung aus der Ferne	262
10.4	Parallele Ein- und Ausgabe	263
10.5	MPI-2	264
10.6	Wird es MPI-3 geben?	264
10.7	Schlusswort	264
Glossar		265
A	Zusammenfassung der MPI-1-Routinen	275
A.1	C-Funktionen	275
A.2	Fortran-Routinen	289
A.3	C++-Methoden	307
B	Die MPI-Implementierung MPICH	321
B.1	Eigenschaften der Modellimplementierung	321
B.1.1	Besonderheiten für den Anwender	321
B.1.2	Portierbarkeit	322
B.1.3	Effizienz	323
B.1.4	Zusätzliche Programme	323
B.1.5	MPICH in der Forschung	323
B.2	Installation und Ausführung der Modellimplementierung	326
B.3	Die Geschichte von MPICH	326
C	Die MPE-Bibliothek	329
C.1	Protokollierung mit MPE	329
C.2	MPE-Graphik	331
C.3	Hilfen in MPE	331

C.4	Das Upshot-Visualisierungssystem	332
D	MPI-Quellen im Internet	337
E	Sprachdetails	339
E.1	Felder in C und Fortran	339
E.1.1	Spalten- und zeilenweise Anordnung	339
E.1.2	Gitter versus Matrizen	340
E.1.3	Höherdimensionale Felder	340
E.2	Aliasing	343
	Literaturverzeichnis	345
	Index	357
	Index für Funktionen, Typen, Datenfelder	363