

Inhaltsverzeichnis

Vorwort	9
Einleitung	11
Für wen ist dieses Buch gedacht	12
Für wen ist dieses Buch nicht gedacht	12
Vorkenntnisse	12
Aufbau dieses Buchs	13
Der ideale Einstiegspunkt für Sie	13
Konventionen und Besonderheiten dieses Buchs	14
Systemanforderungen	14
Beispieldateien zu diesem Buch	15
Danksagung	15
Support	15
Bleiben Sie in Verbindung	15
1 Einführung in die parallele Programmierung	17
Multicore-Programmierung	19
Mehrfache Befehlsströme, mehrfache Datenströme	19
Multithreading	20
Synchronisierung	22
Speedup	23
Das Amdahlsche Gesetz	23
Gustafsons Gesetz	25
Entwurfsmuster	26
Gleichzeitigkeit erkennen	28
Algorithmische Struktur	31
Unterstützende Strukturen	32
Zusammenfassung	32
Schnellreferenz	33
2 Aufgabenparallelität	35
Einführung	36
Threads	37
Die Klasse Task	38
Func<TResult>-Delegaten verwenden	44

Unbehandelte Ausnahmen in Aufgaben	47
Fallbeispiel: Sortieren	53
Bubblesort	53
Einfügesortieren	54
Pivotsortieren	54
Verwendung der Barrier-Klasse	55
Überarbeitung der Pivotsortierung	58
Aufgaben abbrechen	59
Ein Beispiel	61
Beziehungen zwischen Aufgaben	62
Fortsetzungsaufgaben	62
Eltern/Kind-Aufgaben	68
Die Work-Stealing-Warteschlange	70
Zusammenfassung	72
Schnellreferenz	73
3 Datenparallelität	75
Sequenzielle Schleifen in parallele Aufgaben umwandeln	77
Wann lohnt sich die Parallelisierung?	80
Die parallele for-Schleife	81
Schleifen abbrechen	84
Ausnahmen abfangen	88
Abhängigkeiten	90
Reduktion	90
Verwendung des MapReduce-Entwurfsmusters	95
Beispiel: Worthäufigkeiten	99
Zusammenfassung	102
Schnellreferenz	102
4 PLINQ	103
Einführung in LINQ	105
PLINQ	108
PLINQ-Operatoren und -Methoden	113
Der ForAll-Operator	113
ParallelExecutionMode	115
WithMergeOptions	116
AsSequential	116
AsOrdered	117
WithDegreeOfParallelism	118
Ausnahmebehandlung	119
Abbruch	121

Reduktion	122
Verwendung von MapReduce in PLINQ	126
Zusammenfassung	129
Schnellreferenz	130
5 Gleichzeitige Auflistungen	131
Allgemeine Konzepte	134
Das Producer-Consumer-Paradigma (Erzeuger-Verbraucher)	134
Low-Level-Synchronisierung	135
SpinLock	135
SpinWait	137
ConcurrentStack	138
ConcurrentQueue	143
ConcurrentBag	143
ConcurrentDictionary	147
BlockingCollection	150
Zusammenfassung	156
Schnellreferenz	157
6 Eigene Implementierungen	159
Gelegenheiten für eigene Implementierungen erkennen	160
Eigene Producer-Consumer-Auflistungen	161
Aufgaben-Partitionierer	169
Fortgeschrittene Anwendungen für eigene Partitionierer	174
Partitioner<TSource>	175
OrderablePartitioner<TSource>	180
Eigene Taskplaner	183
Der Kontext-Taskplaner	184
Der Taskplaner	185
Zusammenfassung	190
Schnellreferenz	191
7 Berichte und Debugging	193
Debuggen mit Visual Studio 2010	194
Livedebugging	195
Post-Mortem-Analysen durchführen	197
Threads debuggen	198
Mit dem Fenster Parallele Aufgaben arbeiten	201
Mit dem Fenster Parallele Stapel arbeiten	204
Die Threadansicht	204
Die Aufgabenansicht	208

Mit dem Concurrency Visualizer arbeiten	209
Die Ansicht CPU-Auslastung	213
Die Ansicht Threads	214
Die Ansicht Kerne	217
Die Beispiel-Anwendung	224
Zusammenfassung	227
Schnellreferenz	228
Stichwortverzeichnis	229
Über den Autor	235