
Contents

1	Introduction	1
2	Concepts	7
	Background: Traditional Operating Systems	8
	What Is a Thread?	10
	Kernel Interaction	13
	The Value of Using Threads	17
	What Kind of Programs to Thread?	24
	What About Shared Memory?	27
	Threads Standards	27
	Performance	28
3	Foundations	33
	Implementation vs. Specification	34
	Thread Libraries.	34
	The Process Structure	35
	Lightweight Processes	37
	Solaris Multithreaded Model.	40
	System Calls	40
	Signals.	43

4	Lifecycle	47
	Thread Lifecycle	48
5	Scheduling	61
	Different Models of Kernel Scheduling	62
	Thread Scheduling	64
	Context Switching	71
	When Should You Care About Scheduling?	80
6	Synchronization	81
	Synchronization Issues	82
	Synchronization Variables	84
	A Stoppable Producer/Consumer Example	101
7	Complexities	105
	Complex Locking Primitives	106
	Other Synchronization Variables	116
	Message Queues	119
	Cross-Process Synchronization Variables	120
	Initialization and Destruction	122
	Synchronization Problems	123
8	TSD	129
	Thread-Specific Data	130
	Global Variables, Constants, and Cheating	135
9	Cancellation	137
	What Cancellation Is	138
	A Cancellation Example	142
	Using Cancellation	148
	Cancellation Safety	154
	Simple Polling	154
10	Signals	157
	Signals in UNIX	158
	The Solaris Implementation of Signal Handling	161
	Don't Use Signal Handlers!	163

11	Details	169
	Attribute Objects	170
	POSIX Thread IDs	177
	Win32 Thread IDs and Thread Handles	179
	Initializing Your Data: pthread_once()	179
	POSIX Namespace Restrictions	181
	Return Values and Error Reporting	181
	Constants Comments	184
	Pthread Futures	186
	Pthread Extensions	187
	Comparing the OS/2, Win32, and POSIX Libraries	189
12	Libraries	191
	The Threads Library	192
	Multithreaded Kernels	192
	Are Libraries Safe?	194
	Stub Functions in libc	201
	New Semantics for System Calls	202
	Are Libraries Safe?	204
	Threads Debugger Interface	204
	Mixing Solaris Pthreads and UI Threads	205
	Comparisons of Different Implementations	205
13	Design	207
	Making Libraries Safe and Hot	208
	Manipulating Lists	213
	Program Design	223
14	Languages	231
	C	232
	C++	232
	Java	233
	Fortran	236
	Ada	236
	Pascal	237

	Smalltalk	237
	Lisp	237
	Eiffel	238
	Commercial Products	238
	Public Pthreads Implementations	242
15	Tools	245
	Static Lock Analyzer	246
	Using a Thread-Aware, Graphical Debugger	246
	Debug Mutexes	249
	Proctool	251
	TNFview	251
16	Performance	255
	Optimization: Objectives and Objections	256
	CPU Time, I/O Time, Contention, Etc.	259
	Limits on Speedup	262
	Benchmarks and Repeatable Testing	268
	The Lessons of NFS	275
17	Hardware	279
	Types of Multiprocessors	280
	Bus Architectures	283
	Memory Systems	293
18	Examples	297
	Threads and Windows	298
	Socket Server (Master/Slave Version)	304
	Socket Server (Producer/Consumer Version)	310
	Other Programs on the Web	312

Appendices

A	Internet	313
	Threads Newsgroup	313
	Code Examples	314

	Vendor's Threads Pages	314
	Threads Research	314
	The SPILT Package for UI/POSIX Compatibility	314
	Freeware Tools	315
	Other Pointers	315
	The Authors on the Net	315
B	Books	317
	Threads Books	317
	Related Books	319
C	Timings	321
D	Mistakes	325
E	APIs	331
	Function Descriptions	332
	Pthread Functions	333
	Pthread Attributes Objects	336
	POSIX Realtime Scheduling	339
	Mutexes	342
	Mutex Attributes Objects	345
	Condition Variables	347
	Condition Variable Attributes Objects	350
	Cancellation Functions	351
	Thread-Specific Data Functions	353
	Semaphores	355
	Signal Functions	359
	Stdio	361
	Glossary	363
	Index	377