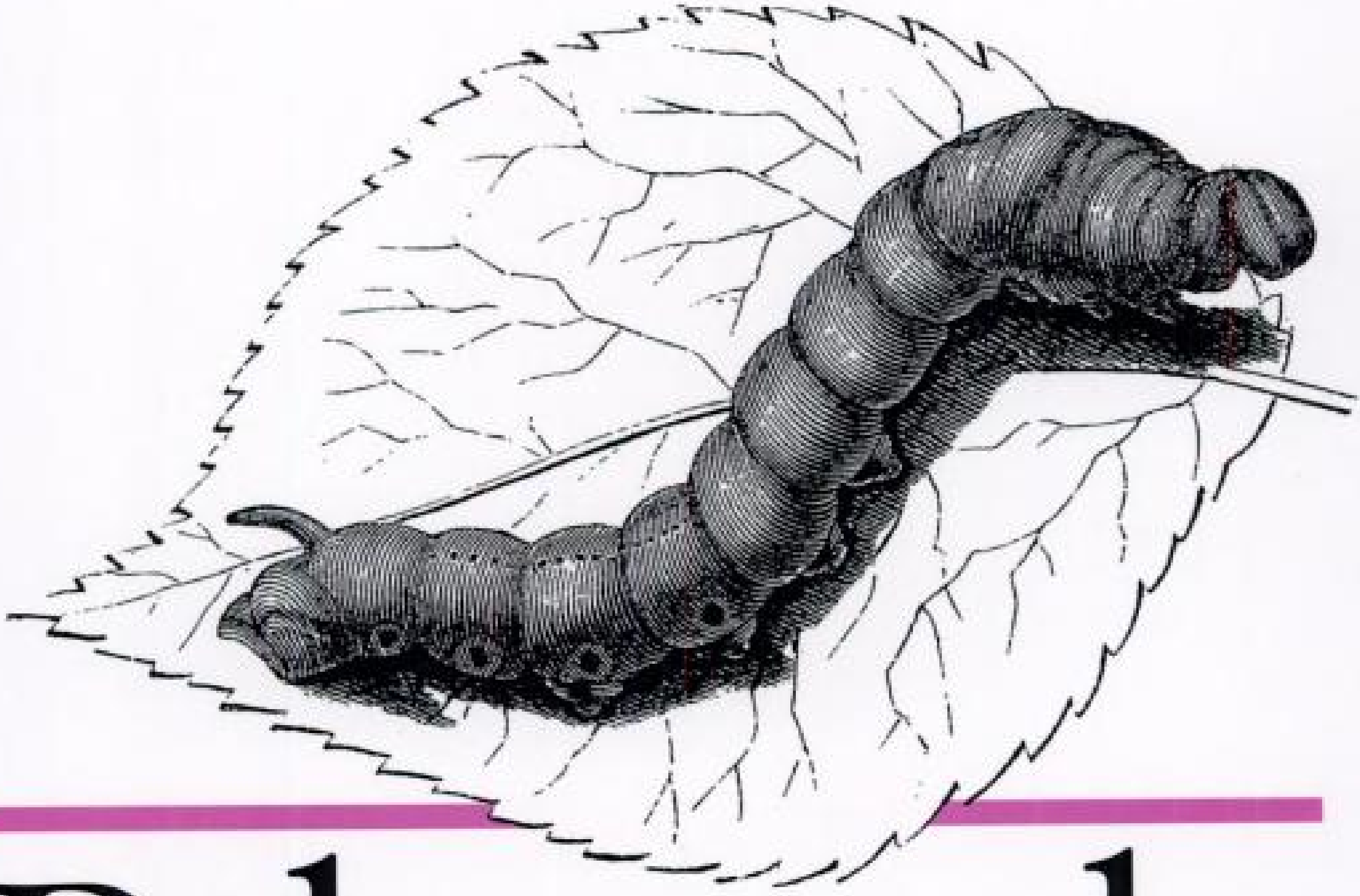
*A POSIX Standard for Better Multiprocessing*



# Pthreads

## *Programming*

A NUTSHELL® HANDBOOK

*Bradford Nichols, Dick Buttlar & Jacqueline Proulx Farrell*

O'Reilly & Associates, Inc.

PThreads Programming: A POSIX Standard for Better Multiprocessing, Dick Buttlar, Jacqueline Farrell, Bradford Nichols, O'Reilly Media, Inc., 1996, 1449364748, 9781449364748, 286 pages. Computers are just as busy as the rest of us nowadays. They have lots of tasks to do at once, and need some cleverness to get them all done at the same time. That's why threads are seen more and more often as a new model for programming. Threads have been available for some time. The Mach operating system, the Distributed Computer Environment (DCE), and Windows NT all feature threads. One advantage of most UNIX implementations, as well as DCE, is that they conform to a recently ratified POSIX standard (originally 1003.4a, now 1003.1c), which allows your programs to be portable between them. POSIX threads are commonly known as pthreads, after the word that starts all the names of the function calls. The standard is supported by Solaris, OSF/1, AIX, and several other UNIX-based operating systems. The idea behind threads programming is to have multiple tasks running concurrently within the same program. They can share a single CPU as processes do, or take advantage of multiple CPUs when available. In either case, they provide a clean way to divide the tasks of a program while sharing data. A window interface can read input on dozens of different buttons, each responsible for a separate task. A network server has to accept simultaneous calls from many clients, providing each with reasonable response time. A multiprocessor runs a number-crunching program on several CPUs at once, combining the results when all are done. All these kinds of applications can benefit from threads. In this book you will learn not only what the pthread calls are, but when it is a good idea to use threads and how to make them efficient (which is the whole reason for using threads in the first place). The authors delves into performance issues, comparing threads to processes, contrasting kernel threads to user threads, and showing how to measure speed. He also describes in a simple, clear manner what all the advanced features are for, and how threads interact with the rest of the UNIX system. Topics include: Basic design techniques Mutexes, conditions, and specialized synchronization techniques Scheduling, priorities, and other real-time issues  Cancellation  UNIX libraries and re-entrant routines  Signals  Debugging tips  Measuring performance  Special considerations for the Distributed Computing Environment (DCE).

DOWNLOAD HERE

Solaris multithreaded programming guide , SunSoft (Firm), 1995, Computers, 158 pages. Multithreading separates a process into many independent execution threads which can improve application responsiveness, program structure, and performance. These threads allow ....

Multicore Application Programming For Windows, Linux, and Oracle Solaris, Darryl Gove, 2010, Computers, 441 pages. Multicore Application Programming is a comprehensive, practical guide to high-performance multicore programming that any experienced developer can use. Author Darryl Gove ....

C++ XML , Fabio Arciniegas, 2002, Computers, 316 pages. Annotation Master the integration of C++ and XML. Written by an XML community leader! Archiniegas writes for www.xml.com, a well-known Website that helps to solve real-world ....

PROFESSIONAL MULTICORE PROGRAMMING DESIGN AND IMPLEMENTATION FOR C++ DEVELOPERS , Cameron Hughes, Jan 1, 2008, , 648 pages. Market_Desc: Ð’Â· Experienced programmers Special Features: Ð’Â· Multicore processors are expected to supplant current microchip technologies by 2009-2010, but there is almost no ....

Threads primer a guide to multithreaded programming, Bil Lewis, Daniel J. Berg, 1996, Computers, 319 pages. Providing an overview of the Solaris and POSIX multithreading architectures, this book explains threads at a level that is completely accessible to programmers and system ....

Learning Python , Mark Lutz, Sep 16, 2009, Computers, 1216 pages. Google and YouTube use Python because it's highly adaptable, easy to maintain, and allows for rapid development. If you want to write high-quality, efficient code that's easily ....

Programming under Mach , Joseph Boykin, 1993, Computers, 490 pages. The book provides the detailed information necessary to write practical programs under Mach. It shows applications writers and programmers how to create programs with multiple ....

Parallel and Distributed Programming Using C++ , Cameron Hughes, Tracey Hughes, 2004, Computers, 691 pages. This book takes complicated parallel programming techniques and presents them in an understandable manner. This title is a no-nonsense tool in the hands of developers and ....

Programming with POSIX Threads , David R. Butenhof, 1997, Computers, 381 pages. Software -- Operating Systems..

Regular Expression Pocket Reference Regular Expressions for Perl, Ruby, PHP, Python, C, Java and .NET, Tony Stubblebine, Jul 18, 2007, Computers, 128 pages. This handy little book offers programmers a complete overview of the syntax and semantics of regular expressions that are at the heart of every text-processing application ....

Programming with threads , Steve Kleiman, Devang Shah, Bart Smaalders, 1996, Computers, 534 pages. A practical guide and reference to developing multithreaded programs on UNIX systems written by the foremost experts on the technology. Covers the two main UNIX threads and the ....