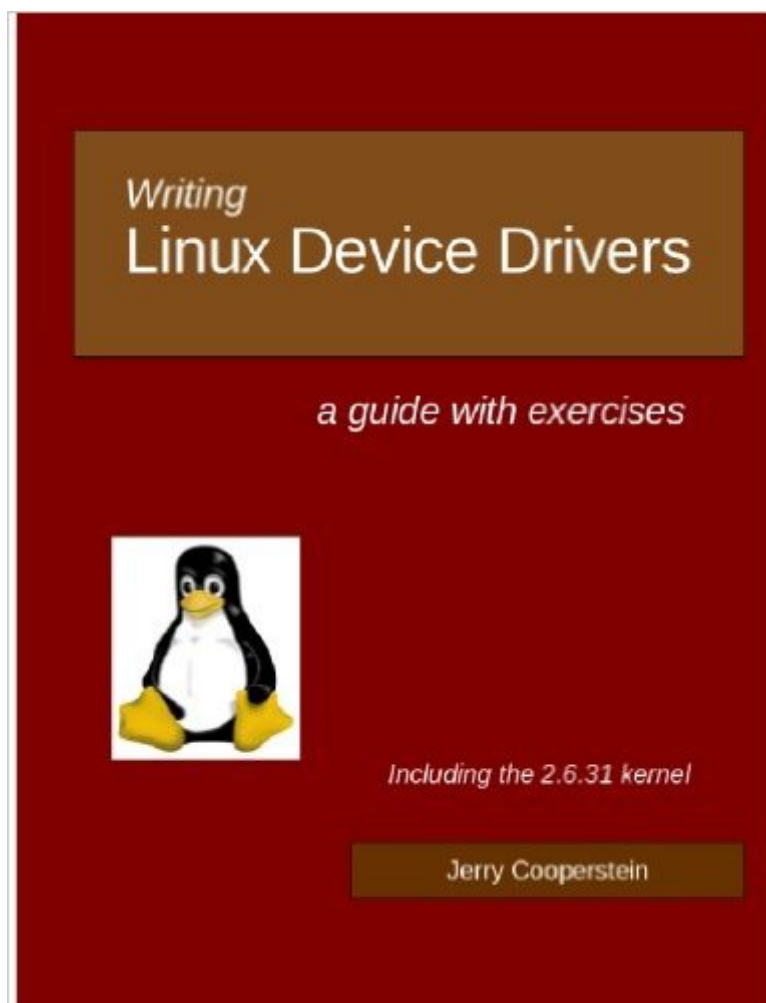# Writing Linux Device Drivers: A Guide With Exercises

Synopsis

Writing Linux Device Drivers is designed to show experienced programmers how to develop device drivers for Linux systems, and give them a basic understanding and familiarity with the Linux kernel. Upon mastering this material, you will be familiar with the different kinds of device drivers used under Linux, and know the appropriate API's through which devices (both hard and soft) interface with the kernel. The purpose is to get you into coding as quickly as possible. Thus we'll tell you early on how to dynamically allocate memory in the simplest way, so you can actually write code, and then later cover the subject more thoroughly. Each section has exercises, most of which involve writing code, designed to help you gain familiarity with programming for the Linux kernel. Solutions are provided. We are not aiming for an expert audience, but instead for a competent and motivated one.

Book Information

Paperback: 394 pages

Publisher: CreateSpace Independent Publishing Platform (September 3, 2009)

Language: English

ISBN-10: 1448672384

ISBN-13: 978-1448672387

Product Dimensions: 7.4 x 0.9 x 9.7 inches

Shipping Weight: 1.9 pounds (View shipping rates and policies)

Average Customer Review: 3.8 out of 5 stars Â See all reviews Â (13 customer reviews)

Best Sellers Rank: #1,023,018 in Books (See Top 100 in Books)   #17 in Â Books > Computers & Technology > Programming > APIs & Operating Environments > Device Drivers   #220 in Â Books > Computers & Technology > Operating Systems > Linux > Programming   #2725 in Â Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Software Development

Customer Reviews

This book was required for one of my Computer Science classes. It does a good job at introducing all the kernel parts, but doesn't go into great detail about any one. It's good for people that are just getting into kernel programming, but I got though the class by using Google and other references that are already out there.

This book reads like a course outline used to teach a course. There are 35 chapters, most about 5 -

10 pages each. It begins by discussing driver issues, and devotes one long chapter (18 pages!) to character drivers. Then the book launches into virtually every kernel programming issue such as interrupts, timers, scheduling, ioctls, etc. Yes, these are useful topics for a device driver developer, but I have already seen most of these topics. The author seems to completely lose sight of the goal of this book: Writing Linux Device Drivers. Finally, in Chapter 24 the author gets back to device drivers and does provide 4 chapters on Network Drivers, and one on USB drivers. Block drivers aren't discussed until the very last chapter (9 pages including exercises).This book can be useful as an outline, a guide to direct your online research (read, Google). But I very much doubt that anyone could write a meaningful device driver using just this book. If you have access to this book, use it as a study guide. But I wouldn't recommend buying it.

I liked this book a lot. It seems that it could just as easily be a (somewhat informal) textbook in a classroom setting or a desk-side companion for a kernel driver hacker. The exercises are very much a part of what makes this book so fun. The writing style and content are a welcomed departure from the standard into-tech-slopping or disjointed glitzy spin noise associated with so many other books these days.The content is very logically presented. It is straight-forward and solid. Every book has issues, but this one shines head and shoulders above the rest, IMO. While it naturally tends toward front-to-rear reading, it doesn't require it the way so many other books do. You can actually jump around rather free-spirited without the essential pre-reading dictated by those books that tell you in chapter one that what they just mentioned will be discussed later in chapter 8, but that [insert next 19 topics] won't be discussed until [add forward reference] because *then* something presented will start to make sense. What the heck?!Seriously, this book is a great book for someone who knows C programming and isn't still looking for the 'any' key on the keyboard. It is very focused on the core information and details of writing Linux drivers as kernel drivers and loadable modules. The examples and the exercises are worth the price of admission, but you get a "right-length" engaging conversational road-trip with the author for free.The cover art should tell you that this book is more about content and less about fluff, which is true. It is a pleasure to be able to recommend a book as fun to read as this one.

Dr. Cooperstein,I just wanted to thank you for your excellent book (and lab solutions manual/code) "Writing Linux Device Drivers." I'm in the process of writing multiple kernel modules as part of my thesis, and I've been having a pretty rough time trying to use existing examples or other research given the significant changes in kernel code. Most of the other books I've found either skirt around

the issues that seem clearly presented in your book...or they explain them only as clearly as the source code itself.Other books that do explain things well often present code that will not work with current linux kernels, and it is very tough for someone trying to learn the concepts to adapt obsolete source code...if I knew how to update the code, I wouldn't need to book in the first place. I think I can say this confidently, as I own just about every book about the linux kernel or linux kernel drivers from O'Reilly, Wrox, Novell, Prentice Hall, Addison Wesley, Osborne, and probably a couple I can't think of right now.Your book is not only well written (and timely for me), but both its content and sample code are directly usable in current linux kernels. I greatly appreciate the simple fact that all the driver code compiles cleanly; that is truly a unique feature at this time...and one that means a lot to someone trying to figure it out. If the online source code continues to be kept to-date with current kernels, your books will be an enduring asset!Thank you for publishing your books.

If you want to quickly learn the basics of building LDD, then go for it. It has everything that you need, nothing that you want.I bought this book after reading through "Essential Linux Device Drivers" by Sreekrishnan Venkateswaran that has so much info but not structured or explained well and had me lost and confused about the basics.

It's a pity i have bought this book. It almost duplicates "Linux Device Drivers, Third Edition" book, which you can get for free from it's author's site. It proposes itself to mirror 2.6.31 kernel changes, but it is not so. Some topics a covered very bad. Read Linux Device Drivers first, and then look at this book - may be you'll find 1 or 2 usefull chapters here. 23$ stupidly lost.I've also came across "Essential Linux Device Drivers" by Sreekrishnan Venkateswaran which I can recomend.

Download to continue reading...

Writing Linux Device Drivers: a guide with exercises Writing Linux Device Drivers: Lab Solutions: a guide with exercises Linux: Linux Command Line - A Complete Introduction To The Linux Operating System And Command Line (With Pics) (Unix, Linux kemel, Linux command line, ... CSS, C++, Java, PHP, Excel, code) (Volume 1) Linux Device Drivers, 3rd Edition Linux Device Drivers, 2nd Edition Practical Linux Programming: Device Drivers, Embedded systems, and the Internet (with CD- ROM) (Programming Series) Writing OpenVMS Alpha Device Drivers in C: Developer's Guide and Reference Manual Writing Windows Device Drivers Writing Device Drivers Writing Device Drivers: Tutorial and Reference Writing Windows Device Drivers Course Notes... Writing DOS Device Drivers in C Writing Os/2 Device Drivers Writing Device Drivers for Sco Unix: A Practical Approach LINUX: Easy Linux For Beginners, Your Step-By-Step Guide To Learning The Linux

Operating System And Command Line (Linux Series) Writing: A Guide Revealing The Best Ways To Make Money Writing (Writing, Writing Skills, Writing Prompts Book 1) CompTIA Linux+ Powered by Linux Professional Institute Study Guide: Exam LX0-103 and Exam LX0-104 (Comptia Linux + Study Guide) Linux For Beginners: The Ultimate Guide To The Linux Operating System & Linux Linux Administration: The Linux Operating System and Command Line Guide for Linux Administrators How to Write a Song: Beginner's Guide to Writing a Song in 60 Minutes or Less (Songwriting, Writing better lyrics, Writing melodies, Songwriting exercises Book 1)