Operating Systems
Three Easy Pieces

REMZI H. ARPACI-DUSSEAU

Remzi H. Arpaci-Dusseau is a Full Professor in the Computer Sciences Department at the University of Wisconsin-Madison. He co-leads a group with his wife, Professor Andrea Arpaci-Dusseau. They have graduated 19 PhD students in their time at Wisconsin, won nine Best Paper awards, and some of their innovations now ship in commercial systems and are used daily by millions of people. Remzi has won the SACM Student Choice Professor of the Year award four times, the Carolyn Rosner "Excellent Educator" award, and the UW-Madison Chancellor's Distinguished Teaching award. Chapters from a freely available OS book he and Andrea co-wrote, found at http://www.ostep.org, have been downloaded millions of times in the past few years. remzi@cs.wisc.edu

Back in 2007, a student (call him Student #1) approached me and asked a simple question: "Do I have to buy the book for this course, or can I get by without it?" The course was undergraduate operating systems (called "CS 537" at the University of Wisconsin-Madison). In teaching the course, I mostly relied upon notes developed by myself and my colleague Andrea Arpaci-Dusseau, and I still thought it was useful for students to have something to read outside of class. So I pushed back a little. "Yes, you should. It's good for you to have another source for the material. Why don't you want to buy it?" The student looked at me sheepishly and said, "Well, I, um, can't really afford it." The book we were using cost over $100, as do many textbooks today.

It was just a small moment, but it led to a big change in how we teach the OS course here at Wisconsin. Although I didn't know it at the time, that simple, honest, and slightly heartbreaking comment led to the creation of a free online operating systems textbook called *Operating Systems: Three Easy Pieces* (sometimes called *OSTEP* and available at http://ostep.org). Chapters of the book have been downloaded millions of times over the past few years, and hundreds of teachers at various colleges and universities have told me they are using the book in their classes.

In the rest of this article, I'll first provide a little more history on how the book developed, discuss how the book is organized, and make the more general case for free online textbooks; indeed, I have a strong belief that all textbooks should be made freely available online. I'll also discuss some keys to success with such an endeavor and present my thoughts on how publishers might evolve in such a free-textbook world.

History

After the encounter with Student #1, I made a snap decision for the course. "No one is required to buy a book for this course. Just come to class, take notes, and that will be enough. Everything you need to know we will cover in class," I declared. The students smiled. For at least one day, I was their $100 hero!

But then along came Student #2 and another encounter I will never forget. After a few classes in this first "no book" semester, this student said, "Professor, I sometimes miss class. For example, tomorrow, I have an interview that I couldn't move. And the other day, I overslept—the class is a bit early in the day for me. So I don't know what to do for those days that I miss and have no notes. And I don't really know anyone else to borrow them from." Now I was a little surprised at one of these comments—the class started at 1:00 p.m. that semester. Undergrads! But the general point hit home: I needed to go beyond the "take notes" approach and provide more material for them.

And thus I hit upon a simple idea. I usually leave the hour or so after class open to wind down. Why not put this hour to good use in service of the class? So after each class, if I had the energy, I would close my door and just write down, in simple text form, what I had just lectured upon in class. Just after class is a great time to do this work: the ideas are fresh in your head and it is relatively easy to write them down.

SYSTEMS

I then posted these crude “text-based lecture captures” to the class Web site. If you’re interested, you can look at them here: http://pages.cs.wisc.edu/~remzi/Classes/537/Fall2008/notes.html.

Honestly, if we’re going to remain friends, you probably shouldn’t look these over—they’re a little embarrassing. Just plain text, no real figures (just some ASCII art), and really very primitive writings.

Then a funny third thing happened: students started to give me (unsolicited) feedback on the writing. And, perhaps a little surprisingly, they were quite positive! In the world of academics, you get a lot of feedback on the work you do, and much of it is negative—those of you who have ever submitted a paper to a conference understand what I am talking about. This positive feedback was a bit like a drug; I wanted more! And so I started to plot how to take these rough notes and make them into something better. And that’s what I have tried to do each semester I taught the class since that time.

Interestingly enough, many upgrades to the book were driven by student feedback. One student wished there were some better diagrams and included detailed notes to me on where to place them on each page, so I spent some time converting ASCII figures into actual EPS graphics. Another said that the raw text was a little hard on the eyes, so I started to typeset each chapter in LaTeX. Some students asked how they could obtain a print copy, which led me to self-publish the book on Lulu.com; we have sold thousands of print copies of a book that is available entirely for free online.

Finally, one student said he would buy a copy if I made a decent cover (the cover at that time: pure black, just text). I am a sucker for a sale, so I asked him, “What would you put on it?” He suggested something cool, like a dinosaur. I had to tell him that the prehistoric beast idea was already taken, but it gave me an idea, and soon enough I had a comet flying across the cover. We know what comets can do to dinosaurs, right?

As a result of all of this effort, we are nearing the completion of what we call a “version 1.0.” The results can be seen at our Web site, http://ostep.org.

Organizing a Book

One major question we had in putting a book together was how to organize the material. Of course, you could just have 10–12 chapters and follow the organization of most other OS books, but that seemed less than interesting. So we started to think about different ways of organizing the material into a few major conceptual themes, and then divide these into short chapters that roughly matched a lecture or half-lecture on a particular topic.

While teaching from different textbooks, we noticed that most books introduced threads and processes early on, and thus soon had to present all thread-related topics, including locks, condition variables, race conditions, and so forth—all very detailed and hard material, and all very early in the semester. However, when we taught the material in this manner, it didn’t quite seem to work; students didn’t even yet understand what an address space was, and we were telling them about the differences between processes (each of which has its own address space) and threads (which share one address space). So we decided to try something different.

At this point, the idea arose to organize the course into three major conceptual pieces: **virtualization** (which covers CPU and memory virtualization), **concurrency** (which introduces threads, locks, condition variables, and related topics), and finally **persistence** (which covers storage devices and file systems). Within each section, we have a lot of short chapters, each on one subtopic (e.g., introduction to CPU scheduling, TLBs, or crash consistency in file systems). While this is a little different from other books, we’ve found that students make more sense of the material in this order. And, in this manner, the title of the book became obvious.

The other advantage of this organization is that it places storage systems (our research specialty) on equal footing with the other parts of the material. Some other books relegate file systems and storage to the very last chapters and thus (in our opinion) spend too much time on virtualization and concurrency at the cost of understanding this important subsystem. After all, what is more important than remembering information for the long term?

One other difference within our approach is that we tend to emphasize mechanism (and the nuts and bolts of how things work) more than policy. This decision stems from a personal belief that learning new policies is relatively easy, but understanding the machinery of systems is hard; class should thus emphasize the hard stuff and leave the easier things for students to learn later.

Why Textbooks Should Be Freely Available Online

There are many reasons textbooks should be made freely available online. Here is a list of some of the big ones:

- **It's the best way to share information with the most people.** Authors spend so much time creating these books; why trap the information inside the standard publishing wall? A casual reader is not going to drop $150 for a book with a few things inside it they are interested in. Making chapters freely available for download allows for casual usage among a much broader group of people.
- **It enables new usage models.** No professor would (likely) dare make a student buy four (expensive) books in order to use a few chapters from each. When book chapters are available online for free, this type of new model is readily available. A more competitive market for specialized sub-books could also arise.
- **It avoids needless revisions.** Authors are currently forced to do a number of silly things because of the way textbook sales work. If the author does not upgrade the book, students happily purchase used copies for very little cost; the publishers, unsurprisingly, are not happy with this, and thus essentially force authors to keep making revision after revision. With no such business model in place, material will get upgraded as needed.
- **It enables chance discovery.** Students find resources today by using search engine tools to browse the Internet or by poking around Wikipedia pages. Having book chapters available for free on the Internet makes chance discovery more likely and possible.
- **It's free.** Making a book free makes it accessible to anyone, regardless of their financial circumstances (assuming they have access to the Internet). If we wish to teach the world, we should make as much information available as inexpensively as possible to as many people as possible.

Keys to Success

In doing this work, I've tried to think about what was essential to realizing some level of success with writing one's own book. Here I list some of these tips for aspiring authors:

- **Develop a class first.** A class (for me) is just 30 lectures, telling one big story (e.g., what is an operating system?) and a number of smaller stories (e.g., what are virtualization, concurrency, persistence?). After being here for some time, Andrea and I had taught the course repeatedly and refined the message each time we rotated through. By the end of this development, we had a pretty good idea of what we wanted to say and how we wanted to say it. Once you have gone through a class a few times, writing it all down is much easier.
- **Improve something each time you teach.** I found the task of writing a book daunting—it's a lot of work! But writing a little now and then didn't sound too bad, and I enjoyed it. They say that the perfect is the enemy of the good, so I just embrace the fact that although the book will never be perfect, I can make it a little better each time. This also gives me a new focus each time I teach the class, which actually makes teaching the same class more interesting than usual.
- **Make each chapter a separate downloadable unit.** There are many reasons to do so and three particularly important ones. First, students won't get overwhelmed by a massive 800-page beast; each chapter, in contrast, is usually short (say 10–20 pages) and thus much less daunting and easier to digest. Second, short chapters enable better discovery via search engine and other related means. A person might search for "semaphores," and it is much easier to then find the exact chapter instead of searching through a book on operating systems; similarly, a Wikipedia page on multi-level page tables can point directly to the right chapter instead of vaguely to an entire book. Third, parts of the book can be used instead of the whole; a professor at another institution can pick and choose chapters from different sources, which would be much harder to do if the entire book is the only unit of usage.
- **Create homework assignments that are reusable.** Book chapters need homework questions to enable students to test their own knowledge. The thought of writing some fixed questions, and then having to update questions regularly, was a non-starter. As a result, we started using an idea we saw in Hennessy and Patterson's *Computer Architecture: A Quantitative Approach*, which was to create computer programs that can generate an infinite number of variants to a certain class of question. In our case, these programs are essentially little simulators that mimic some aspect of an OS. For example, a virtual memory simulator might generate a particular configuration (physical memory of size X, a Y-bit virtual address space) and then ask you to translate certain addresses from virtual to physical. By adding more simulators over time, you give students a richer, more interactive way to quiz themselves about the material.
- **Be responsive to feedback.** We actively encourage feedback from anyone who reads the book and credit them for any fix or update that arises from their suggestions. Many students have thus found typos for us, which we have fixed; many professors and instructors have suggested more substantial changes, which we have implemented as well. While we can't accommodate every request, we read each one carefully and then decide what to do. In all cases, we get back to the suggester as quickly as we can.
- **Realize you don't have to cover everything.** One last point about making a book: it doesn't have to be a bible. Use it to spark a student's interest and cover most important topics, especially topics you care about. It's OK if not everything is covered in a textbook; rather, what you are giving students is a way to under-

stand the major pieces and the ability (hopefully) to be able to fill details in themselves at a later time.

Of course, none of these suggestions are useful without a fair amount of hard work, for which there is little substitute. But, if done right, the work is rewarding and spread out, and you get lots of thanks from people around the world.

Aside: Why Free Doesn't Mean Open Source

People often say to me, "That's great you're doing a free book. Why isn't it on GitHub so I can hack on it, too?" My reaction to that is usually, "Uh, no thanks." Why so unfriendly, you ask?

The answer: we strongly believe that a book should have a single voice. This voice communicates one coherent body of knowledge to the reader. If each chapter were written by different people, this voice would likely be lost and the experience lessened. Whatever the model of collaboration, we believe that the important thing is that the author or group of authors work hard to maintain that single voice.

What This Means for Publishers

Probably the biggest change that will occur, should all textbooks become free, is to the world of publishers, who will find that their services (in the current form) are not much needed. However, they could save themselves by doing a number of things.

First, publishers should split out their services and offer parts of said services to authors. For example, publishers could help with marketing and advertising of free textbooks. In addition, publishers could offer editorial services as a separate service. Even printing could be split off and offered (although they are behind here, thanks to Lulu.com and other similar services). Instead of going with one publisher for all of these things, an author could pick and choose what he or she needs.

Second, publishers should figure out more ways to publish print copies at low cost. I've spoken with publishers who said they want to do low-cost books, and then turn around and say they can't do a book for less than $50 or $60. In contrast, at Lulu.com, you can print single copies of a book on demand for $20 to $30. Publishers need to get their costs down and become competitive in offering low-cost print books. Students still like print, and by selling both digital and print at low cost and high volume, publishers could still make money. There seems to be some recalcitrance in the industry that prevents this.

Conclusion

I strongly believe that textbooks should be free. *OSTEP* is just one such book, and is and will always be freely available online and at a low cost in print forms. But *OSTEP* is just one book. There needs to be more! If you are a teacher of a class, think about what it would take to convert your own personal lecture notes into something more widely shared. Soon, you might have a textbook on your hands, and the free textbook revolution can truly begin!