



Why Software Sucks...and What You Can Do About It

By David S. Platt

[Download now](#)

[Read Online](#) 

Why Software Sucks...and What You Can Do About It By David S. Platt

“I’ve just finished reading the best computer book [*Why Software Sucks...*] since I last re-read one of mine and I wanted to pass along the good word. . . . Put this one on your must-have list if you have software, love software, hate programmers, or even ARE a programmer, because Mr. Platt (who teaches programming) has set out to puncture the bloated egos of all those who think that just because they can write a program, they can make it easy to use. . . . This book is funny, but it is also an important wake-up call for software companies that want to reduce the size of their customer support bills. If you were ever stuck for an answer to the question, ‘Why do good programmers make such awful software?’ this book holds the answer.”

—John McCormick, *Locksmith columnist, TechRepublic.com*

“I must say first, I don’t get many computing manuscripts that make me laugh out loud. Between the laughs, Dave Platt delivers some very interesting insight and perspective, all in a lucid and engaging style. I don’t get much of that either!”

—Henry Leitner, *assistant dean for information technology and senior lecturer on computer science, Harvard University*

“A riotous book for all of us downtrodden computer users, written in language that we understand.”

—Stacy Baratelli, *author’s barber*

“David’s unique take on the problems that bedevil software creation made me think about the process in new ways. If you care about the quality of the software you create or use, read this book.”

—Dave Chappell, *principal, Chappell & Associates*

“I began to read it in my office but stopped before I reached the bottom of the first page. I couldn’t keep a grin off my face! I’ll enjoy it after I go back home and find a safe place to read.”

—Tsukasa Makino, *IT manager*

“David explains, in terms that my mother-in-law can understand, why the software we use today can be so frustrating, even dangerous at times, and gives us some real ideas on what we can do about it.”

—*Jim Brosseau, Clarrus Consulting Group, Inc.*

A Book for Anyone Who Uses a Computer Today...and Just Wants to Scream!

Today’s software sucks. There’s no other good way to say it. It’s unsafe, allowing criminal programs to creep through the Internet wires into our very bedrooms. It’s unreliable, crashing when we need it most, wiping out hours or days of work with no way to get it back. And it’s hard to use, requiring large amounts of head-banging to figure out the simplest operations.

It’s no secret that software sucks. You know that from personal experience, whether you use computers for work or personal tasks. In this book, programming insider David Platt explains why that’s the case and, more importantly, why it doesn’t have to be that way. And he explains it in plain, jargon-free English that’s a joy to read, using real-world examples with which you’re already familiar. In the end, he suggests what you, as a typical user, without a technical background, can do about this sad state of our software—how you, as an informed consumer, don’t have to take the abuse that bad software dishes out.

As you might expect from the book’s title, Dave’s expose is laced with humor—sometimes outrageous, but always dead on. You’ll laugh out loud as you recall incidents with your own software that made you cry. You’ll slap your thigh with the same hand that so often pounded your computer desk and wished it was a bad programmer’s face. But Dave hasn’t written this book just for laughs. He’s written it to give long-overdue voice to your own discovery—that software does, indeed, suck, but it shouldn’t.



[Download Why Software Sucks...and What You Can Do About It ...pdf](#)



[Read Online Why Software Sucks...and What You Can Do About It ...pdf](#)

Why Software Sucks...and What You Can Do About It

By David S. Platt

Why Software Sucks...and What You Can Do About It By David S. Platt

“I’ve just finished reading the best computer book [*Why Software Sucks...*] since I last re-read one of mine and I wanted to pass along the good word. . . . Put this one on your must-have list if you have software, love software, hate programmers, or even ARE a programmer, because Mr. Platt (who teaches programming) has set out to puncture the bloated egos of all those who think that just because they can write a program, they can make it easy to use. . . . This book is funny, but it is also an important wake-up call for software companies that want to reduce the size of their customer support bills. If you were ever stuck for an answer to the question, ‘Why do good programmers make such awful software?’ this book holds the answer.”

—John McCormick, *Locksmith columnist, TechRepublic.com*

“I must say first, I don’t get many computing manuscripts that make me laugh out loud. Between the laughs, Dave Platt delivers some very interesting insight and perspective, all in a lucid and engaging style. I don’t get much of that either!”

—Henry Leitner, *assistant dean for information technology and senior lecturer on computer science, Harvard University*

“A riotous book for all of us downtrodden computer users, written in language that we understand.”

—Stacy Baratelli, *author’s barber*

“David’s unique take on the problems that bedevil software creation made me think about the process in new ways. If you care about the quality of the software you create or use, read this book.”

—Dave Chappell, *principal, Chappell & Associates*

“I began to read it in my office but stopped before I reached the bottom of the first page. I couldn’t keep a grin off my face! I’ll enjoy it after I go back home and find a safe place to read.”

—Tsukasa Makino, *IT manager*

“David explains, in terms that my mother-in-law can understand, why the software we use today can be so frustrating, even dangerous at times, and gives us some real ideas on what we can do about it.”

—Jim Brosseau, *Clarrus Consulting Group, Inc.*

A Book for Anyone Who Uses a Computer Today...and Just Wants to Scream!

Today’s software sucks. There’s no other good way to say it. It’s unsafe, allowing criminal programs to creep through the Internet wires into our very bedrooms. It’s unreliable, crashing when we need it most, wiping out hours or days of work with no way to get it back. And it’s hard to use, requiring large amounts of head-banging to figure out the simplest operations.

It’s no secret that software sucks. You know that from personal experience, whether you use computers for

work or personal tasks. In this book, programming insider David Platt explains why that's the case and, more importantly, why it doesn't have to be that way. And he explains it in plain, jargon-free English that's a joy to read, using real-world examples with which you're already familiar. In the end, he suggests what you, as a typical user, without a technical background, can do about this sad state of our software—how you, as an informed consumer, don't have to take the abuse that bad software dishes out.

As you might expect from the book's title, Dave's expose is laced with humor—sometimes outrageous, but always dead on. You'll laugh out loud as you recall incidents with your own software that made you cry. You'll slap your thigh with the same hand that so often pounded your computer desk and wished it was a bad programmer's face. But Dave hasn't written this book just for laughs. He's written it to give long-overdue voice to your own discovery—that software does, indeed, suck, but it shouldn't.

Why Software Sucks...and What You Can Do About It By David S. Platt Bibliography

- Rank: #1466269 in Books
- Published on: 2006-10-01
- Released on: 2006-09-21
- Original language: English
- Number of items: 1
- Dimensions: 8.90" h x .70" w x 6.00" l, .98 pounds
- Binding: Paperback
- 272 pages

 [Download Why Software Sucks...and What You Can Do About It ...pdf](#)

 [Read Online Why Software Sucks...and What You Can Do About I ...pdf](#)

Download and Read Free Online Why Software Sucks...and What You Can Do About It By David S. Platt

Editorial Review

From the Back Cover

“I’ve just finished reading the best computer book [*Why Software Sucks...*] since I last re-read one of mine and I wanted to pass along the good word. . . . Put this one on your must-have list if you have software, love software, hate programmers, or even ARE a programmer, because Mr. Platt (who teaches programming) has set out to puncture the bloated egos of all those who think that just because they can write a program, they can make it easy to use. . . . This book is funny, but it is also an important wake-up call for software companies that want to reduce the size of their customer support bills. If you were ever stuck for an answer to the question, ‘Why do good programmers make such awful software?’ this book holds the answer.”

—John McCormick, *Locksmith columnist, TechRepublic.com*

“I must say first, I don’t get many computing manuscripts that make me laugh out loud. Between the laughs, Dave Platt delivers some very interesting insight and perspective, all in a lucid and engaging style. I don’t get much of that either!”

—Henry Leitner, *assistant dean for information technology and senior lecturer on computer science, Harvard University*

“A riotous book for all of us downtrodden computer users, written in language that we understand.”

—Stacy Baratelli, *author’s barber*

“David’s unique take on the problems that bedevil software creation made me think about the process in new ways. If you care about the quality of the software you create or use, read this book.”

—Dave Chappell, *principal, Chappell & Associates*

“I began to read it in my office but stopped before I reached the bottom of the first page. I couldn’t keep a grin off my face! I’ll enjoy it after I go back home and find a safe place to read.”

—Tsukasa Makino, *IT manager*

“David explains, in terms that my mother-in-law can understand, why the software we use today can be so frustrating, even dangerous at times, and gives us some real ideas on what we can do about it.”

—Jim Brosseau, *Clarrus Consulting Group, Inc.*

A Book for Anyone Who Uses a Computer Today...and Just Wants to Scream!

Today’s software sucks. There’s no other good way to say it. It’s unsafe, allowing criminal programs to creep through the Internet wires into our very bedrooms. It’s unreliable, crashing when we need it most, wiping out hours or days of work with no way to get it back. And it’s hard to use, requiring large amounts of head-banging to figure out the simplest operations.

It’s no secret that software sucks. You know that from personal experience, whether you use computers for work or personal tasks. In this book, programming insider David Platt explains why that’s the case and, more importantly, why it doesn’t have to be that way. And he explains it in plain, jargon-free English that’s a joy

to read, using real-world examples with which you're already familiar. In the end, he suggests what you, as a typical user, without a technical background, can do about this sad state of our software—how you, as an informed consumer, don't have to take the abuse that bad software dishes out.

As you might expect from the book's title, Dave's exposé is laced with humor—sometimes outrageous, but always dead on. You'll laugh out loud as you recall incidents with your own software that made you cry. You'll slap your thigh with the same hand that so often pounded your computer desk and wished it was a bad programmer's face. But Dave hasn't written this book just for laughs. He's written it to give long-overdue voice to your own discovery—that software does, indeed, suck, but it shouldn't.

About the Author

David S. Platt runs Rolling Thunder Computing (www.rollthunder.com), an education and consulting practice. He has more than twenty years of experience as a programmer, teaches software development at Harvard University Extension School and at companies all over the world, and is a popular speaker at conferences. He is the author of nine previous books—including *Introducing Microsoft .NET, Third Edition*, *The Microsoft Platform Ahead*, and *Understanding COM+* (all Microsoft Press)—as well as many journal articles and newsletters. In 2002, Microsoft designated him a Software Legend. Dave lives in Ipswich, Massachusetts.

Excerpt. © Reprinted by permission. All rights reserved.

Today's software sucks. There's no other good way to say it. It's unsafe, allowing criminal programs to creep through the Internet wires into our very bedrooms. It's unreliable, crashing when we need it most, wiping out hours or days of work with no way to get it back. And it's hard to use, requiring large amounts of head banging to figure out the simplest operations.

This isn't news to you, is it? You're not dumb, contrary to what that yellow-and-black book series would have you believe. Today's software really does suck, as you've always thought. That's why you laughed when you read the title of this book. And it wasn't a voluntary laugh, either, was it? You didn't read the title and think to yourself, "Hey, that's kinda funny, I think I'll laugh." No. The title touched a nerve, like a smell knifing through your cerebral cortex to your reptilian midbrain, and you burst out with an involuntary, explosive laugh. When you started thinking again, you probably thought something like, "Hey, cool! Here's a guy (an articulate, credentialed, handsome, and modest one, too, by the look of things) telling it like it is, saying in print what I've always thought. About time, too." Indeed it is. This book explains in nontechnical, jargon-free language exactly how this situation came about, and what you can do to change it. "Damn right I'll buy it! And copies for my friends, too." Excellent idea. Told you that you weren't a dummy, didn't I?

Fifteen years ago—even ten—ordinary people didn't use software in their daily lives. My parents did not send e-mail birthday cards to their granddaughters, or expect me to send them digital photos that way. They balanced their checking accounts on a paper register and maintained their schedules by writing in ink on paper calendars with pretty pictures. The few people who did regularly use software usually had to as part of their jobs—for example, travel agents using airline reservation systems to issue tickets. These people used small, custom-tailored applications that required expensive, proprietary hardware and extensive training and ongoing support and that weren't useful for any other purpose. The World Wide Web was this geeky thing for wooly haired academics. Most people didn't even know it existed, let alone that you could steal music or download dirty pictures from it.

That's changed completely, almost overnight in societal terms, and seemingly without our noticing it. My parents now use a financial software package to manage their checking accounts, which pays bills electronically so that they don't run out of stamps, and automatically shows electronic deposits and cleared

checks as they arrive at the bank. Not only do they want granddaughter photos by e-mail, but they're also asking for live streaming video. With unlimited high-speed access available to home users for about \$40 per month, the Web has accomplished what proponents of nuclear power predicted and never managed to do, which is become too cheap to meter. It's now so ubiquitous that the state of Pennsylvania has removed its motto, "The Keystone State," from its automobile license plates and replaced it with the state Web address, www.state.pa.us. Florida has done Pennsylvania one better, keeping its slogan but replacing the state name with the Web address MyFlorida.com. And the travel agent's profession has been largely wiped out by ordinary people having cheap, easy access to information that once required very expensive and complicated setups.

We live our lives today in a sea of software, but most users have no idea how software comes into being or why it works the way it does. We only know that we don't like it very much. Everyone has a software horror story or five, just as we have airline horror stories about flights delayed for hours on the runway and luggage misrouted to Timbuktu. You worked all day with your program, didn't realize you had to explicitly save your document, then that damn program crashed and you lost it all. And the random pattern of bits on your screen looked exactly like it was flipping you the bird as it went down. From TIME magazine's 1982 pick as "Man of the Year" the personal computer's reputation has gone downhill.

Software doesn't have to suck, and it shouldn't, but it does. One reason is that the programmers and architects and managers who develop it don't understand their customers anywhere near as well as they should, as well as designers in other industries have been forced to. Their products suck not because developers don't know how to do this or that particular thing, but because they don't know which things to do. They're disconnected from their customers (some would call them victims)—the ones who pay their salaries—and they usually don't realize it. They often solve the wrong problem, adding features that no one cares about except themselves, harming all users in the process. If they understood who their users are, they'd make different, better choices.

For example, Microsoft Office applications such as Word and Excel allow a user to drag the main menu (the one with File, Edit, and Help on it) from its usual location at the top of the window to any other edge of the screen, or even floating free over a document (see Figure 4). I've never met or even heard of anybody, and I mean not one single person, who has ever used this feature, not even my contacts on the Office team. So why does it hurt the program to have it? For many reasons. I occasionally reach my mouse up to click the File menu and instead overshoot by a little, usually when I've had too much coffee, and find myself dragging the menu bar around on my screen. I have to stop what I'm doing, drag the bar back where it belongs, dock it in the place where it belongs, above the toolbar; then spend half a minute cursing the ancestry of the morons who wrote this dumb, so-called feature. This might not sound like much, but 30 seconds twice a day, times a billion users, add up to wasting roughly 27 human life spans every single day.¹ I, and the vast majority of users, would be more productive if this feature was completely removed from the program so that we wouldn't waste our time and break our concentration by having to deal with this nonsense. In addition, the extra programming instructions (called "code" by programmers) needed to provide this feature increase the possibility of crashing errors and security vulnerabilities, in the same way as more moving parts on any mechanical device render it less reliable. The world would be a better place if Microsoft had taken the money it spent on designing, writing, testing, debugging, documenting, and supporting this foolishness and burned it, or better yet, given it to me. Worst of all, these antiparadise doodads consume development resources that could have and should have been spent on items that most users actually do care about, such as falling over dead less often and not losing work when that happens. The ultimate example of resources wasted on counterproductive features is Clippy, the talking, dancing, irritating paper clip in Microsoft Office (added in 1997, deactivated by popular demand in 2002).²

Not only does software suck today, but it will continue to do so until we demand that it stop. Our cars

improved in terms of safety (air bags and antilock brakes), reliability (better engineering leading to fewer failures), and usability (CD players and cup holders) only when customers started demanding them, buying cars that had them and passing over those that didn't. My mother always told me not to complain about something unless I had a better idea. I do have many better ideas, and I give them to you as I point out the foolishness of some current software design decisions. While this isn't primarily a how-to book, I reveal all the tricks that I know to mitigate the most awful effects of bad program design—for example, turning off the confirmation dialog ("Are you sure? Really sure? Really, really, really sure?") that appears when you send a document to the Windows Recycle Bin. More importantly, this book explains what you and I need to do to make our voices heard in the software industry so that it stops sucking, hopefully soon. It worked for Clippy, didn't it?

At the same time, I have to tell you that developing software is different in many ways from developing physical objects, things that you can drop on your foot. When you build, say, a table out of wood, the inherent properties of the material preclude some design choices and dictate others. For example, you know that you can't weld it, but you can fasten it with screws, and you can make it only so thin before it bends unacceptably. Software, on the other hand, imposes very few inherent constraints. It is almost infinitely tractable. As Frederick Brooks said in his classic book on software development, *The Mythical Man-Month, Anniversary Edition* (Addison Wesley, 1995)—which I've suggested that he retitle to "The Mystical Geek Week," because of today's accelerated project schedules—programmers work with "nearly pure thought-stuff." I've always described programming as trying to stuff smoke into a bottle. So I'll explain to you what really is inherently difficult to make a program do (surviving a power blip, for example, or making a new program compatible with previous versions³) as opposed to what's simply a boneheaded design decision by an idiot designer who should have known better.

I teach software development at Harvard University Extension School and at companies all over the world. I've written nine books for programmers and development managers, along with many journal articles and newsletters. Despite these disadvantages, I promise you that my writing is never dry. I did my best to stay away from geek jargon. For example, you won't find the word gigabyte appearing in this book,⁴ but I'll refer to "a quarter's worth of disk space," which is about the same thing at the time of this writing.

I've found this book surprisingly easy to write compared to some of my others. The late sportswriter, Red Smith (1906-1982), liked to say, "Writing is easy. You just open your veins and bleed." As I hope you've already seen, I feel passionately about the issues I raise here. When I finish ranting in a live talk ("Windows XP is a decent product for the price. Windows 98 is an inferior piece of crap that isn't worth the powder to blow it to hell"), my students often say, "Gosh, Platt, stop beating around the bush and tell us how you really feel." To which I reply, "If you're accusing me of calling 'em as I see 'em, I plead guilty as charged." This book is not just an explanation for you; it's my cry to the heavens for sanity in this crazy profession. And I hope when you read this, you'll add your voice to mine.

Notes:

1. A billion minutes is 694,444 days, 1,901 years, or just over 27 spans of 70 years.
2. I am not alone in detesting that vile, Gollum-like creature. As lawyers Dahlia Lithwick and Brandt Goldstein wrote in *Me v. Everybody: Absurd Contracts for an Absurd World* (Workman Publishing, 2003): "3. The Maniacal-Paper-Clip-With-Eyebrows Provision. You will delete/disable/destroy whatever it is that allows that inane little bastard to leap around the bottom right-hand corner of my screen, emitting what can only be described as a mechanical fart and incessantly observing: 'I see that you're writing a ransom note . . .' or assuming that I wish it to turn all my letters into spreadsheets and my correspondence into numbered lists."
3. A favorite programmer joke runs, "How did God manage to create the world in only six days? He didn't

have any installed base he had to worry about backwards compatibility with.” My editor points out that He also skimped on documentation.

4. Yes, I know, except for this once, wiseguy.

Users Review

From reader reviews:

Robert Riggio:

As people who live in often the modest era should be update about what going on or data even knowledge to make all of them keep up with the era which is always change and make progress. Some of you maybe will update themselves by reading books. It is a good choice to suit your needs but the problems coming to you actually is you don't know which you should start with. This Why Software Sucks...and What You Can Do About It is our recommendation to cause you to keep up with the world. Why, because this book serves what you want and want in this era.

Sandra Bryson:

Spent a free a chance to be fun activity to try and do! A lot of people spent their leisure time with their family, or their very own friends. Usually they undertaking activity like watching television, going to beach, or picnic from the park. They actually doing same thing every week. Do you feel it? Do you wish to something different to fill your own personal free time/ holiday? Might be reading a book may be option to fill your no cost time/ holiday. The first thing that you will ask may be what kinds of publication that you should read. If you want to attempt look for book, may be the publication untitled Why Software Sucks...and What You Can Do About It can be very good book to read. May be it can be best activity to you.

Jeffrey Blough:

Reading can called mind hangout, why? Because if you find yourself reading a book specially book entitled Why Software Sucks...and What You Can Do About It your head will drift away trough every dimension, wandering in every single aspect that maybe not known for but surely will end up your mind friends. Imaging every single word written in a e-book then become one web form conclusion and explanation in which maybe you never get ahead of. The Why Software Sucks...and What You Can Do About It giving you one more experience more than blown away your head but also giving you useful facts for your better life in this particular era. So now let us show you the relaxing pattern here is your body and mind is going to be pleased when you are finished looking at it, like winning a. Do you want to try this extraordinary shelling out spare time activity?

George Conner:

Many people said that they feel fed up when they reading a book. They are directly felt the item when they get a half regions of the book. You can choose the book Why Software Sucks...and What You Can Do About It to make your own reading is interesting. Your personal skill of reading proficiency is developing when you just like reading. Try to choose easy book to make you enjoy to learn it and mingle the feeling about

book and examining especially. It is to be initial opinion for you to like to wide open a book and learn it. Beside that the book Why Software Sucks...and What You Can Do About It can to be your brand new friend when you're experience alone and confuse using what must you're doing of these time.

Download and Read Online Why Software Sucks...and What You Can Do About It By David S. Platt #M387Y9KDVR0

Read Why Software Sucks...and What You Can Do About It By David S. Platt for online ebook

Why Software Sucks...and What You Can Do About It By David S. Platt Free PDF d0wnl0ad, audio books, books to read, good books to read, cheap books, good books, online books, books online, book reviews epub, read books online, books to read online, online library, greatbooks to read, PDF best books to read, top books to read Why Software Sucks...and What You Can Do About It By David S. Platt books to read online.

Online Why Software Sucks...and What You Can Do About It By David S. Platt ebook PDF download

Why Software Sucks...and What You Can Do About It By David S. Platt Doc

Why Software Sucks...and What You Can Do About It By David S. Platt MobiPocket

Why Software Sucks...and What You Can Do About It By David S. Platt EPub

M387Y9KDVR0: Why Software Sucks...and What You Can Do About It By David S. Platt