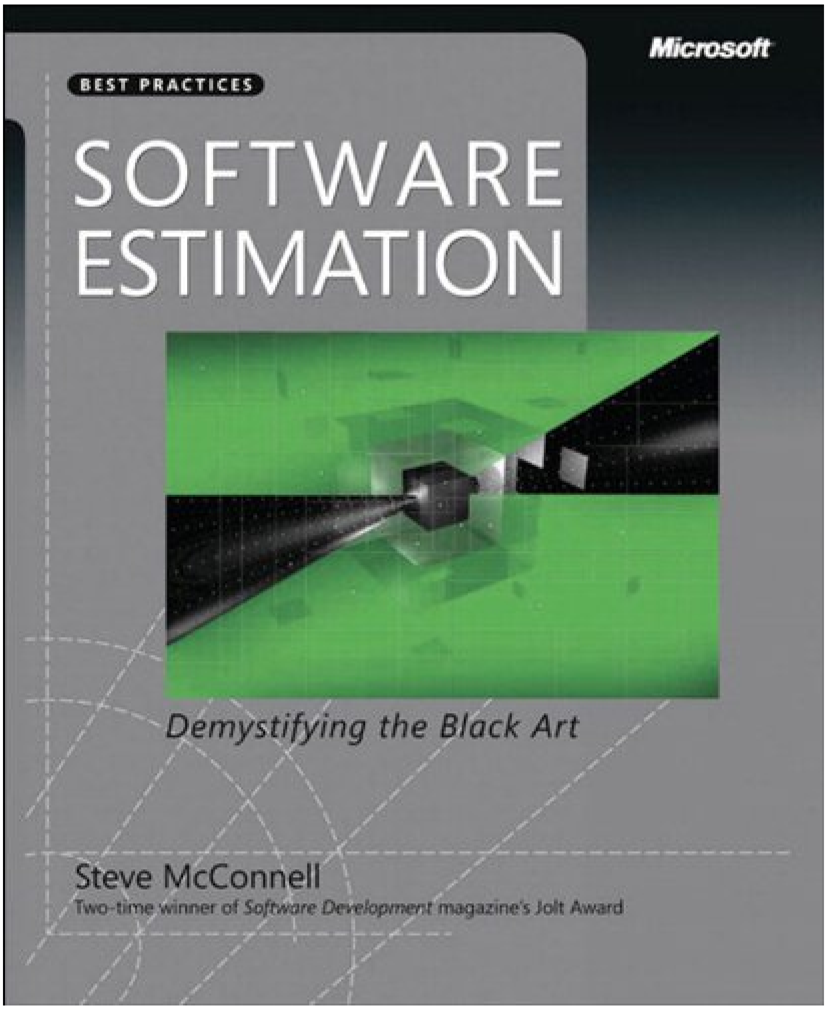
Software Estimation: Demystifying The Black Art (Developer Best Practices)


Microsoft
BEST PRACTICES
SOFTWARE ESTIMATION
Demystifying the Black Art
Steve McConnell
Two-time winner of Software Development magazine's Jolt Award

Synopsis

Often referred to as the â œblack artâ • because of its complexity and uncertainty, software estimation is not as difficult or puzzling as people think. In fact, generating accurate estimates is straightforwardâ ”once you understand the art of creating them. In his highly anticipated book, acclaimed author Steve McConnell unravels the mystery to successful software estimationâ ”distilling academic information and real-world experience into a practical guide for working software professionals. Instead of arcane treatises and rigid modeling techniques, this guide highlights a proven set of procedures, understandable formulas, and heuristics that individuals and development teams can apply to their projects to help achieve estimation proficiency. Discover how to: Estimate schedule and costâ ”or estimate the functionality that can be delivered within a given time frame Avoid common software estimation mistakes Learn estimation techniques for you, your team, and your organization * Estimate specific project activitiesâ ”including development, management, and defect correction Apply estimation approaches to any type of projectâ ”small or large, agile or traditional Navigate the shark-infested political waters that surround project estimates When many corporate software projects are failing, McConnell shows you what works for successful software estimation.

Book Information

File Size: 2500 KB

Print Length: 308 pages

Simultaneous Device Usage: Up to 5 simultaneous devices, per publisher limits

Publisher: Microsoft Press; 1 edition (February 22, 2006)

Publication Date: February 22, 2006

Sold by:Â Digital Services LLC

Language: English

ASIN: B00JDMPOVQ

Text-to-Speech: Enabled

X-Ray: Not Enabled

Word Wise: Not Enabled

Lending: Not Enabled

Enhanced Typesetting: Not Enabled

Best Sellers Rank: #205,694 Paid in Kindle Store (See Top 100 Paid in Kindle Store) #33 inÂ Kindle Store > Kindle eBooks > Computers & Technology > Networking > Client-Server

Systems   #110 inÂ Books > Computers & Technology > Networking & Cloud Computing > Data in the Enterprise > Client-Server Systems   #140 inÂ Kindle Store > Kindle eBooks > Computers & Technology > Operating Systems

Customer Reviews

McConnell immediately differentiates between software estimation as a science and software estimation as an art within the first pages of the introduction to this work - he explains that while software estimation research is currently focused on improving estimation techniques so that project results are achieved within +/- 5% of esimated results, the techniques of which are best included in commerical software estimation tools that are used to pursue the science of estimation for very large projects, the typical software organization struggles to avoid estimates that are incorrect by 100% or more, and Software Estimation discusses the estimation techniques comprising, although not limited to, the art of estimation that will reduce estimation error to about 25% or less, which is what the vast majority of software projects need. This book is absolutely the best software estimation text I have read to date. As Joel Spolsky mentions on the back cover of the book, the vast majority of software project managers still think that estimates are based on multiples of a gut feel. This philosophy completely ignores the vast body of research and hands-on experience of the last few decades. McConnell explains with sufficient depth all of the ways to create "ball park" estimates, and if there is one chapter of this book that you read, Chapter 4 "Where Does Estimation Error Come From?" is the one that I recommend - the explanation of the Cone of Uncertainty is so well explained that anyone can understand it. Although I have never read Code Complete by the same author, I did purchase Rapid Development several years ago - and I must say that McConnell (or at least his editors) has vastly improved his writing skills.

I must admit, I was very surprised to see an excellent source on software development time and cost estimation from Microsoft Press. However, when I saw that the author was Steve McConnell, the author of Code Complete (which I thought was a great book) I knew that this book would have something to offer.His definition of the purpose of software estimation is important: "The primary purpose of software estimation is not to predict a project’s outcome; it is to determine whether a project’s targets are realistic enough to allow the project to be controlled to meet them."From this premise, the author continues to prove that software estimation is about determining if you can control a project to bring it in reasonably close to the organization’s targets. In addition to this, Mr. McConnell’s arguments for overestimation being better than underestimation are rather good and have convinced me to continue using my risk time and budget items that I've been using and recommending to others.In chapter 4, the author identifies four main sources of project estimation errors:-Inaccurate information about the project being estimated-Inaccurate information about the capabilities of the organization that will perform the project-Too much chaos in the project to support accurate estimation (that is, trying to estimate a moving target)-Inaccuracies arising from the estimation process itselfThat last one presents the need for me to give a little more information. The author suggests that bias and subjectivity of the estimator cause major estimation errors in the estimation process. This is proven through exercise case studies of time estimation in training classes.

This is Steve McConnell's latest published effort on software engineering. The book is divided into three parts: estimation concepts, techniques, and challenges respectively. The first part of the book, on concepts, is just a broad overview designed to convince the software engineer of the value of estimation and how, done incorrectly, it can mess up what would have been a successful project. The second part, on techniques, is the meatiest part of the book. Here the author combines tried and true computational techniques with your own judgement and reminds you to get the expert opinion of other people too when estimating a project. Finally, the third and last part of the book, on challenges, talks about how you must think "out of the box" when you confront a software project whose size, effort, and schedule may be hard to quantify completely. Finally, McConnell points out the importance of politics and group dynamics in general in estimation. This is important, as many software engineers and project managers tend to be lone wolves that want to go off in a corner and solve problems by themselves. I highly recommend this book to all software engineers and particularly to software project managers. does not show the table of contents, so I do that here:CRITICAL ESTIMATION CONCEPTS1 What is an "estimate"? 32 How good an estimator are you? 153 Value of accurate estimates 214 Where does estimation error come from?

Download to continue reading...

Software Estimation: Demystifying the Black Art (Developer Best Practices) Detection Estimation and Modulation Theory, Part I: Detection, Estimation, and Filtering Theory MicrosoftÃ Â® Exchange Server 2010 Best Practices (IT Best Practices - Microsoft Press) Writing Secure Code for Windows VistaÃ Â® (Developer Best Practices) The iOS 5 Developer's Cookbook: Core Concepts and Essential Recipes for iOS Programmers (Developer's Library) The Swift Developer's Cookbook (includes Content Update Program) (Developer's Library) Exploring Mac App Development Strategies: Patterns & Best Practices for Clean Software Architecture on the Mac with Swift 2.0 and

Tests ARM System Developer’s Guide: Designing and Optimizing System Software (The Morgan Kaufmann Series in Computer Architecture and Design) Interviewing in Swift: Algorithms and Data Structures: Your guide in helping you prepare for the real world of software engineering interviews as an iOS or Mac OS developer. Being Geek: The Software Developer’s Career Handbook Demystifying The Microchip PIC Microcontroller For Engineering Students: Following The KISS Principle Hot Thai Kitchen: Demystifying Thai Cuisine with Authentic Recipes to Make at Home The Wine Handbook - A Field Guide to Demystifying Wine (The Back To Basics Handbooks) Calling C and C++ from IDL Using Microsoft Visual C++ and call_external: Demystifying creating and debugging an IDL DLL So You Think You Know Washington State Wine? (2016-17): Demystifying the Economics of Wine (Washington Wine Series) Demystifying Anorexia Nervosa: An Optimistic Guide to Understanding and Healing (Developmental Perspectives in Psychiatry) Demystifying Patanjali: The Yoga Sutras: The Wisdom of Paramhansa Yogananda as Presented by his Direct Disciple, Swami Kriyananda Demystifying Patanjali: The Yoga Sutras (Aphorisms): The Wisdom of Paramhansa Yogananda Presented by his direct disciple, Swami Kriyananda Demystifying the Out-of-Body Experience: A Practical Manual for Exploration and Personal Evolution Black Magic Spells: Black Magic Spells for Beginners (Black Magick)