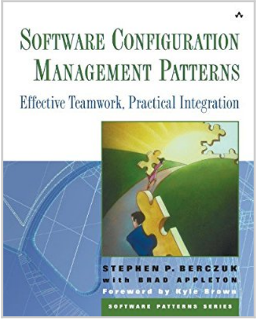
Software Configuration Management Patterns
Effective Teamwork, Practical Integration
Stephen P. Berczuk
with Brad Appleton
Foreword by Kyle Brown
Software Patterns Series

Software Configuration Management Patterns: Effective Teamwork, Practical Integration

By Stephen P. Berczuk, Brad Appleton

Software Configuration Management Patterns: Effective Teamwork, Practical Integration By Stephen P. Berczuk, Brad Appleton

Stereotypes portray software engineers as a reckless lot, and stereotypes paint software configuration management (SCM) devotees as inflexible. Based on these impressions, it is no wonder that projects can be riddled with tension! The truth probably lies somewhere in between these stereotypes, and this book shows how proven SCM practices can foster a healthy team-oriented culture that produces better software. The authors show that workflow, when properly managed, can avert delays, morale problems, and cost overruns. A patterns approach (proven solutions to recurring problems) is outlined so that SCM can be easily applied and successfully leveraged in small to medium sized organizations. The patterns are presented with an emphasis on practicality. The results speak for themselves: improved processes and a motivated workforce that synergize to produce better quality software.

Software Configuration Management Patterns: Effective Teamwork, Practical Integration

By Stephen P. Berczuk, Brad Appleton

Software Configuration Management Patterns: Effective Teamwork, Practical Integration By Stephen P. Berczuk, Brad Appleton

Stereotypes portray software engineers as a reckless lot, and stereotypes paint software configuration management (SCM) devotees as inflexible. Based on these impressions, it is no wonder that projects can be riddled with tension! The truth probably lies somewhere in between these stereotypes, and this book shows how proven SCM practices can foster a healthy team-oriented culture that produces better software. The authors show that workflow, when properly managed, can avert delays, morale problems, and cost overruns. A patterns approach (proven solutions to recurring problems) is outlined so that SCM can be easily applied and successfully leveraged in small to medium sized organizations. The patterns are presented with an emphasis on practicality. The results speak for themselves: improved processes and a motivated workforce that synergize to produce better quality software.

Software Configuration Management Patterns: Effective Teamwork, Practical Integration By Stephen P. Berczuk, Brad Appleton Bibliography

- Sales Rank: #915108 in Books
- Published on: 2003-11-14
- Original language: English
- Number of items: 1
- Dimensions: 9.00" h x .70" w x 7.30" l, .99 pounds
- Binding: Paperback
- 218 pages

Download and Read Free Online Software Configuration Management Patterns: Effective Teamwork, Practical Integration By Stephen P. Berczuk, Brad Appleton

Editorial Review

From the Back Cover

Effective software configuration management (SCM) strategies promote a healthy, team-oriented culture that produces better software. ***Software Configuration Management Patterns*** alleviates software engineers' most common concerns about software configuration management—perceived rigidity and an overemphasis on process.

Through the use of patterns, the authors show that a properly managed workflow can avert delays, morale problems, and cost overruns. The patterns approach illustrates how SCM can be easily and successfully applied in small- to mid-size organizations. By learning how these patterns relate to each other, readers can avoid common mistakes that too often result in frustrated developers and reduced productivity.

Key coverage includes instruction on how to:

Build and foster a development environment focused on producing optimal teamwork and quality products. ***Software Configuration Management Patterns*** also includes a detailed list of SCM tools and thorough explanations of how they can be used to implement the patterns discussed in the book. These proven techniques will assist readers to improve their processes and motivate their workforce to collaborate in the production of higher quality software.

0201741172B09202002 About the Author
Stephen P. Berczuk has been developing object-oriented software applications since 1989, often as part of geographically distributed teams. He has been an active member of the Software Patterns community since the first PLoP conference in 1994, and did early work on the relationship between organization, software architecture, and design patterns. He has an M.S. in Operations Research from Stanford University and an S.B. in Electrical Engineering from MIT.**Brad Appleton** has been a software developer since 1987 and has extensive experience using, developing, and supporting SCM environments for teams of all shapes and sizes. A former Patterns++ section editor for the *C++ Report*, Brad is also well versed in object-oriented design and agile software development, and cofounded the Chicago Patterns and Chicago Agile Development Groups. He holds an M.S. in Software Engineering and a B.S. in Computer Science and Mathematics.

0201741172AB09202002 Excerpt. © Reprinted by permission. All rights reserved.
Software configuration management is not what I do. I am not a software configuration management person. I am not an organizational anthropology person. However, I discovered early on that understanding organizations, software architecture, and configuration management was essential to doing my job as a software developer. I also find this systems perspective on software engineering interesting. I build software systems, and configuration management is a very important and often neglected part of building software systems. In this book, I hope that I can show you how to avoid some of the problems I have encountered so that you can build systems more effectively with your team.I should probably explain what I mean in distinguishing between software configuration management (SCM) people and people who build software systems. The stereotype is that configuration management people are concerned with tools and control. They are conservative, and they prefer slow, predictable progress. They are also "the few" as compared with "the many" developers in an organization. Software engineers (so the stereotype goes) are reckless. They want to build things fast, and they are confident that they can code their way out of any situation. These are extreme

stereotypes, and in my experience, the good software engineers and the good release/quality assurance/configuration management people have a common goal: They are focused on delivering quality systems with the least amount of wasted effort.Good configuration management practice is the not the silver bullet to building systems on time, nor are patterns, Extreme Programming (XP), the Unified Process, or anything else that you might hear about. It is, however, a part of the toolkit that most people ignore because they fear "process," often because of bad experiences in the past (Weigers 2002).This book describes some common software configuration management practices. The book will be particularly interesting to software developers working in small teams who suspect that they are not using software configuration management as effectively as they can. The techniques that we describe are not tool specific. As with any set of patterns or best practices, the ease with which you can apply the patterns may depend on whether your tool explicitly supports them.Why I Wrote This BookI started my software development career with a small R&D group based in the Boston area. Aside from the many interesting technical problems we encountered as part of our jobs, we had the added twist of having joint development projects with a group in our parent company's home base in Rochester, New York. This experience helped me recognize early in my career that software development wasn't just about good design and good coding practices but also about coordination among people in the same group and even teams in different cities. Our group took the lead in setting up the mechanics of how we would share code and other artifacts of the development process. We used the usual things to make working together easier, such as meetings, teleconferences, and e-mail lists. The way we set up our (and the remote team's) software configuration management system to share code played a very large part in making our collaboration easier.The people who set up the SCM process for our Boston group used techniques that seemed to have been tried throughout their careers. As I moved on to other organizations, I was amazed to find how many places were struggling with the same common problems—problems that I knew had good solutions. This was particularly true because I had been with a number of start-ups that were only one or two years old when I joined. One to two years is often the stage in a start-up where you are hiring enough people that coordination and shared vision are difficult goals to attain.A few years into my career, I discovered patterns. Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides were just finishing the book Design Patterns (Gamma et al. 1995), and the Hillside Group was organizing the first Pattern Languages of Programs (PLoP) conference. There is a lot of power in the idea of patterns because they are about using the right solution at the right time and because patterns are interdisciplinary; they are not just about domain- or language-specific coding techniques but about how to build software from all perspectives, from the code to the team. I presented a number of papers in workshops at the various PLoP conferences that dealt with patterns at the intersection of design, coding, and configuration management (Berczuk 1995, 1996a, 1996b; Appleton et al. 1998; Cabrera et al. 1999; Berczuk and Appleton 2000).At one PLoP conference, I met Brad Appleton, who is more an SCM expert than I am. We coauthored a paper about branching patterns (Appleton et al. 1998), just one aspect of SCM. After much encouragement from our peers, we started working on this book. I hope that this book helps you avoid some common mistakes, either by making you aware of these approaches or by providing you with documentation you can use to explain methods that you already know about to others in your organization.Who Should Read This BookI hope that anyone who builds software and uses a configuration management system can learn from this book. The details of the configuration management problem change depending on the types of systems that you are building, the size of the teams, and the environment that you work in. Because it's probably impossible to write a book that will address everyone's needs and keep everyone's interest, I had to limit what I was talking about. This book will be most valuable to someone who is building software, or managing a software project, in a small to medium-size organization where there is not a lot of defined process. If you are in a small company, a start-up, or a small project team in a larger organization, you will benefit most from the lessons in this book. Even if your organization has a very well-defined, heavy process that seems to be impeding progress, you'll be able to use the patterns in this book to focus better on some of the key tasks of SCM.How To Read This BookThe introduction explains some basic concepts of software configuration management and the notation that the diagrams use. Part I provides background information about SCM and patterns.

Chapter 1 introduces the software configuration management concepts used in this book. Chapter 2 talks about some of the forces that influence the decisions you make about what sort of SCM environment you have. Chapter 3 introduces the concept of patterns and the patterns in this book and how they relate to each other. Part II consists of patterns that illustrate problems and solutions to common SCM problems. Chapters 1 and 2 also define the general problems that this book addresses. To understand the how patterns fit together, you should read Chapter 3 to get an overview of the language.After you have read the first three chapters, you can browse the patterns in Part II, starting with one you find interesting and following with ones that relate to your problem. Another approach is to read the patterns in order and form a mental picture of the connections between them.The references to the other patterns in the book appear in the introductory paragraph for each chapter and in the Unresolved Issues section at the end of each chapter, using a presentation like this: Active Development Line (5). The number in parentheses is the chapter number that contains the pattern. Because this is a large field to cover, some of the context and Unresolved Issues sections don't refer to other patterns, either in the book or elsewhere, because they haven't been documented as of this writing. In this case, you will see a description about what a pattern might cover.Origins of This MaterialMuch of the material in this book has its origins in papers written for various Pattern Languages of Programs conferences by me, Brad Appleton, Ralph Cabrera, and Robert Orenstein. The patterns have been greatly revised from the original material, but it's appropriate to mention these papers to acknowledge the roles of others in this work: "Streamed Lines: Branching Patterns for Parallel Software Development" (Appleton et al. 1998), "Software Reconstruction: Patterns for Reproducing the Build" (Cabrera et al. 1999), "Configuration Management Patterns" (Berczuk 1996b).—***Steve Berczuk***

Arlington, Massachusetts, June 2002
steve@berczuk.com
http://www.berczuk.com

Why I CoWrote This Book with SteveI began my software development career in 1987 as a part-time software tools developer to pay for my last year of college. Somehow it "stuck" because I've been doing some form of tool development ever since (particularly SCM tools), even when it wasn't my primary job. I even worked (briefly) for a commercial SCM tool vendor, and part of my job was to stay current on the competition. So I amassed as much knowledge as I could about other SCM tools on the market. Even after I changed jobs, I continued my SCM pursuits and frequented various tool user groups on the Internet.At one time, I longed to advance the state of the art in SCM environments and kept up with all the latest research. I soon became frustrated with the vast gap between the "state of the art" and the "state of the practice." I concluded that I could do more good by helping advance the state of the practice to use available tools better. Not long after that, I discovered software patterns and the patterns community. It was clear that these folks were onto something important in their combination of analysis and storytelling for disseminating recurring best practices of software design.At the time, hardly anyone in the design patterns community was attempting to write SCM patterns. SCM is, after all, the "plumbing of software development" to a lot of programmers: Everyone acknowledges that they need it, but no one wants to have to dive into it too deeply and get their hands entrenched in it. They just want it to work and not to have to bother with it all that much.It didn't take long for me to hook up with Steve Berczuk. We wrote several SCM patterns papers together (with Ralph Cabrera) as part of my ACME project at http://acme.bradapp.net/ and later decided to work on this book. We hope this small but cohesive set of core patterns about integration and teamwork helps the unsuspecting developers-cum-project-lead survive and thrive in successfully leading and coordinating their team's collaborative efforts and integrating the results into working software.—***Brad Appleton***

Arlington Heights, Illinois, June 2002
brad@bradapp.net
http://www.bradapp.net

0201741172P10072002 Users Review**From reader reviews:**

Angela Rodriguez:This Software Configuration Management Patterns: Effective Teamwork, Practical Integration are reliable for you who want to certainly be a successful person, why. The reason why of this Software Configuration Management Patterns: Effective Teamwork, Practical Integration can be one of the great books you must have will be giving you more than just simple studying food but feed an individual with information that maybe will shock your preceding knowledge. This book is handy, you can bring it just about everywhere and whenever your conditions at e-book and printed types. Beside that this Software Configuration Management Patterns: Effective Teamwork, Practical Integration giving you an enormous of experience for instance rich vocabulary, giving you tryout of critical thinking that we know it useful in your day action. So , let's have it appreciate reading.

Eleanor Hayes:Software Configuration Management Patterns: Effective Teamwork, Practical Integration can be one of your starter books that are good idea. We recommend that straight away because this guide has good vocabulary that may increase your knowledge in vocabulary, easy to understand, bit entertaining but still delivering the information. The article author giving his/her effort that will put every word into satisfaction arrangement in writing Software Configuration Management Patterns: Effective Teamwork, Practical Integration but doesn't forget the main stage, giving the reader the hottest as well as based confirm resource info that maybe you can be certainly one of it. This great information can certainly drawn you into fresh stage of crucial contemplating.

Jose Campbell:The book untitled Software Configuration Management Patterns: Effective Teamwork, Practical Integration contain a lot of information on the idea. The writer explains the girl idea with easy technique. The language is very clear and understandable all the people, so do certainly not worry, you can easy to read the idea. The book was compiled by famous author. The author brings you in the new time of literary works. You can read this book because you can read on your smart phone, or gadget, so you can read the book throughout anywhere and anytime. If you want to buy the e-book, you can start their official web-site along with order it. Have a nice examine.

Bryant Davidson:In this age globalization it is important to someone to acquire information. The information will make someone to understand the condition of the world. The health of the world makes the information easier to share. You can find a lot of recommendations to get information example: internet, newspapers, book, and soon. You can observe that now, a lot of publisher in which print many kinds of book. The book that recommended to you is Software Configuration Management Patterns: Effective Teamwork, Practical Integration this e-book consist a lot of the information on the condition of this world now. This kind of book was represented how does the world has grown up. The words styles that writer use for explain it is easy to understand. Often the writer made some analysis when he makes this book. That's why this book acceptable all of you.

Download and Read Online Software Configuration Management Patterns: Effective Teamwork, Practical Integration By Stephen P. Berczuk, Brad Appleton #80RJYLC4WNM

Read Software Configuration Management Patterns: Effective Teamwork, Practical Integration By Stephen P. Berczuk, Brad Appleton for online ebookSoftware Configuration Management Patterns: Effective Teamwork, Practical Integration By Stephen P. Berczuk, Brad Appleton 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 Software Configuration Management Patterns: Effective Teamwork, Practical Integration By Stephen P. Berczuk, Brad Appleton books to read online.Online Software Configuration Management Patterns: Effective Teamwork, Practical Integration By Stephen P. Berczuk, Brad Appleton ebook PDF downloadSoftware Configuration Management Patterns: Effective Teamwork, Practical Integration By Stephen P. Berczuk, Brad Appleton DocSoftware Configuration Management Patterns: Effective Teamwork, Practical Integration By Stephen P. Berczuk, Brad Appleton MobipocketSoftware Configuration Management Patterns: Effective Teamwork, Practical Integration By Stephen P. Berczuk, Brad Appleton EPub80RJYLC4WNM: Software Configuration Management Patterns: Effective Teamwork, Practical Integration By Stephen P. Berczuk, Brad Appleton