国外优秀信息科学与技术系列教学用书

CMM 实现指南

——如何改进软件过程

（影印版）

CMM IMPLEMENTATION GUIDE

Choreographing Software Process Improvement

■ Kim Caputo

高等教育出版社
Higher Education Press
Pearson Education
出版集团

国外优秀信息科学与技术系列教学用书

CMM 实现指南

——如何改进软件过程

（影印版）

CMM IMPLEMENTATION GUIDE

Choreographing Software Process Improvement

Kim Caputo

高 等 教 育 出 版 社

Pearson Education 出版集团

图字：01-2002-3779 号

CMM Implementation Guide: Choreographing Software Process Improvement, First Edition

Kim Caputo

图书在版编目(CIP)数据

CMM 实现指南:如何改进软件过程/(美)卡普托(Caputo,K). —影印本. —北京:高等教育出版社,2002.10
ISBN 7-04-011401-1

Ⅰ.C… Ⅱ.卡… Ⅲ.软件工程—高等学校—教材—英文 Ⅳ.TP311.5

中国版本图书馆 CIP 数据核字(2002)第 081682 号

CMM实现指南——如何改进软件过程(影印版)

Kim Caputo

出版发行	高等教育出版社	**购书热线**	010-64054588
社　　址	北京市东城区沙滩后街 55 号	**免费咨询**	800-810-0598
邮政编码	100009	**网　　址**	http://www.hep.edu.cn
传　　真	010-64014048		http://www.hep.com.cn
经　　销	新华书店北京发行所		
印　　刷	北京民族印刷厂		
开　　本	787×1092　1/16	**版　　次**	2002 年 10 月第 1 版
印　　张	21.25	**印　　次**	2002 年 10 月第 1 次印刷
字　　数	520 000	**定　　价**	32.00 元(含光盘)

本书如有缺页、倒页、脱页等质量问题，请到所购图书销售部门联系调换。

出 版 说 明

20 世纪末，以计算机和通信技术为代表的信息科学和技术对世界经济、科技、军事、教育和文化等产生了深刻影响。信息科学技术的迅速普及和应用，带动了世界范围信息产业的蓬勃发展，为许多国家带来了丰厚的回报。

进入 21 世纪，尤其随着我国加入 WTO，信息产业的国际竞争将更加激烈。我国信息产业虽然在 20 世纪末取得了迅猛发展，但与发达国家相比，甚至与印度、爱尔兰等国家相比，还有很大差距。国家信息化的发展速度和信息产业的国际竞争能力，最终都将取决于信息科学技术人才的质量和数量。引进国外信息科学和技术优秀教材，在有条件的学校推动开展英语授课或双语教学，是教育部为加快培养大批高质量的信息技术人才采取的一项重要举措。

为此，教育部要求由高等教育出版社首先开展信息科学和技术教材的引进试点工作。同时提出了两点要求，一是要高水平，二是要低价格。在高等教育出版社和信息科学技术引进教材专家组的努力下，经过比较短的时间，第一批由教育部高等教育司推荐的 20 多种引进教材已经陆续出版。这套教材出版后受到了广泛的好评，其中有不少是世界信息科学技术领域著名专家、教授的经典之作和反映信息科学技术最新进展的优秀作品，代表了目前世界信息科学技术教育的一流水平，而且价格也是最优惠的，与国内同类自编教材相当。这套教材基本覆盖了计算机科学与技术专业的课程体系，体现了权威性、系统性、先进性和经济性等特点。

目前，教育部正在全国 35 所高校推动示范性软件学院的建设，这也是加快培养信息科学技术人才的重要举措之一。为配合软件学院的教学工作，结合各软件学院的教学计划和课程设置，高等教育出版社近期聘请有关专家和软件学院的教师遴选推荐了一批相应的原版教学用书，正陆续组织出版，以方便各软件学院开展双语教学。

我们希望这些教学用书的引进出版，对于提高我国高等学校信息科学技术的教学水平，缩小与国际先进水平的差距，加快培养一大批具有国际竞争力的高质量信息技术人才，起到积极的推动作用。同时我们也欢迎广大教师和专家们对我们的教材引进工作提出宝贵的意见和建议。联系方式：hep.cs@263.net。

高等教育出版社

二〇〇二年九月

Foreword

Many, perhaps most, software projects have historically been late, over budget, and have had less functionality than expected, causing serious quality concerns. At the same time, software has become pervasive in modern society, and the power and complexity of software-dependent systems has exploded. Although we have many more powerful tools and methodologies today than ever before, it seems that the demand outstrips our ability to address software issues effectively.

Addressing the software challenge has led to many initiatives, including the application of Total Quality Management concepts to improving the processes for developing and maintaining software. Much of this work has been based on The Capability Maturity Model (CMM) developed by the Software Engineering Institute at Carnegie Mellon University. The Software CMM is a useful and powerful tool developed to provide guidance in software process improvement efforts. It is a generally applicable reference model that focuses on "what" rather than how.

Over the last decade, a network of process specialists has grown to exchange ideas and lessons learned about effective approaches to implementing software process improvement. Ultimately, improvement means changing the behaviors of organizations—a difficult task. Over a thousand professionals working on software process improvement from around the world meet annually at the Software Engineering Process Group (SEPG) National Conference. At the local level, Software Process Improvement Networks (SPINs) meet regularly to discuss improvement issues.

This book captures the lessons learned in real-world improvement efforts in a form accessible to everyone, many of which may have otherwise limited opportunity to learn from others. The author's guidance in using the CMM for defining, deploying, and improving processes will help many to attack software process improvement more effectively. The choreography metaphor should offer additional insight.

Software process improvement is not rocket science, but its relative simplicity does not mean it's easy. Caputo's *CMM Implementation Guide* is a useful resource to consider when embarking on the improvement journey.

Mark Paulk
Software Engineering Institute
Pittsburgh, PA
February 1998

Preface

Industry and Community

The software industry has experienced tremendous and unprecedented growth over the past 40 years, and our global and economic infrastructures are becoming more and more dependent on software. The software industry has been one of the fastest growing industries ever to exist. Along with it, a supporting industry has developed and grown into a strong international community: the community of SEPGs.

What Is an SEPG?

A Software Engineering Process Group (SEPG) is a team of software professionals with responsibility for driving and facilitating software process improvement efforts within a software organization.

This book was written by an SEPG member for SEPG members, with the intention of capturing the knowledge, tools, and techniques that we have found to be critical to our work. In this book, I have captured the things I wish somebody had told me when I first started in the SEPG business.

Goals of This Book

This book provides many thought-provoking ideas that are intended to help you be more effective in your software process improvement efforts. This book takes Humphrey's *Managing the Software Process,* Fowler and Rifkin's *Software Engineering Process Group Guide,* and Software Engineering Institute's *Capability Maturity Model®* to a higher level of abstraction, a deeper level of understanding, and a broader level of application to build a foundation for a disciplined approach to software process improvement.

The goals of this book are as follows:

- To help SEPGs understand the image of successful improvement as represented by the guidelines in the Capability Maturity Model® for Software (CMM®) and explain concepts in insightful ways that will help you to make sense of what to do.
- To help SEPGs work as partners with their customers (the people in the software organizations: the managers and project teams).
- To help SEPGs understand how to drive and facilitate process improvement efforts to implement processes that their organizations are willing to accept and follow.
- To provide tools and techniques for SEPGs to use for driving and facilitating process improvements that lead to successful implementation efforts.
- To provide encouragement for starting and continuously optimizing software process improvement efforts.

Learning from Experience

At Unisys Corporation, when our first SEPG was formed there was not much guidance on how to do the work of an SEPG. There was a lot of information out there, but it took us two to three years to read it and translate it to our own situation. We had information about *what* to do, but it was difficult to derive the practical methods for *how* to make organization-wide software process improvements. We had to learn the hard way—from experience.

Who should read this book?

- If you are currently part of an SEPG you can benefit from these ideas, which encourage working toward continuous optimization. You can benefit from seeing the relationship between your experience and ours, recognizing a repeatable pattern, and focusing your improvement efforts to save time and effort. I expect this book to help existing SEPGs to be more effective and possibly more creative.
- If you are just starting an SEPG you probably don't have the luxury of a two- or three-year start-up phase. You can't afford to learn the hard way, but you can learn from the

experience of others. You can benefit from our experience to save time and effort. By packaging this information and integrating it with practical experience, I expect this book to help new SEPGs to start their efforts and gain momentum.

- If your organization does not have an SEPG and has no plans to staff one but you are willing to champion improvement efforts, you might find these tools helpful. By starting a few small efforts and making progress, you will gain some short-term benefits, and you will begin to build an awareness of the focus that is involved in organization-wide software process improvement efforts. I expect this book to help process improvement champions to make the improvements that they can make on their own, but there may be global improvements that they cannot make on their own. If the champions want to convince senior management of the need for an SEPG and want to explain what the SEPG members would do for the organization, this book will be helpful.

A Choreographic Perspective

As I considered the source of inspiration for my work in software process improvement, I found a most unlikely source in my early life experience with dance and choreography. The analogy comes alive in this book. Dance is like producing software. Dancers perform the steps of the dance; software programmers perform the steps of the software process. Learning a dance could involve things such as reading the notation and interpreting the steps, or it could involve watching someone who has performed the steps before, trying to do the steps, trying the steps with the music, and making changes when something isn't working. Improving a process involves similar activities. As professionals, programmers are gifted and talented individuals, brilliant in their own right, who can collaborate to give a performance that does not diminish any individual performance. At least, this is our hope; sometimes the chemistry is wonderful, and sometimes it isn't.

Choreography involves movement of the body, guiding one or more dancers through certain steps and through changing rhythms while maintaining balance to create a peak performance for their audience.

Software process improvement involves the movement of an organization, guiding one or more individuals through certain activities and through changing conditions while maintaining balance to create a peak performance for their customers.

In both cases there is a need to understand how to move people's energy to continuously strive for excellence. Just as art is not done simply for the sake of art itself, but for the sake of enlightenment, entertainment, and encouragement of the audience, so software process improvement is not done simply for the sake of software process improvement itself, but for the sake of the customers whose lives are becoming increasingly dependent on software. We must learn to improve and create excellence for their sake.

How to Use This Book

Because this book is based on both theoretical models and practical experience, as you use the models you will encounter many of the same experiences described here. Some things may not apply to you right now, but they may apply to you later.

How to Use the Files on the CD-ROM

The CD-ROM contains templates, samples, and presentation materials that you can use to get started more quickly. However, please note that these materials are not intended for you to use directly but rather are intended for you to adapt to fit your organization. These tools will reduce your start-up time, but you must add your own finishing touches if you hope to succeed.

I am not suggesting that these materials are "silver bullets" or perfect templates. They fit the situation at the time they were used. If your situation is different, the materials won't fit as well. And your situation is sure to be different: a different day with different people (and remember that even the same people might be different on a different day).

These materials might be helpful to people who find it easier to start with something and make changes rather than start with nothing and create their own materials from scratch. There is a phenomenon called IKIWISI, an acronymn for "I'll know it when I see it." We often can't describe what we want, but we will know what we want when we see it. We will also know what we don't want when we see it. Both reactions provide valuable information for producing successful results. These templates give you something to look at so that you can consider what you want and what you don't want. Then you can create exactly what you need.

Acknowledgments

I gratefully acknowledge the following people who have had a significant influence on me and the writing of this book.

The choreographers whom I had the privilege to watch in rehearsals when I was young. They didn't know I was in the wings learning from them. Anthony Tudor, Eugene Loring, Agnes de Mille, Murray Louis, and Bela Lewitzky.

The choreographers and coaches who worked with me personally. They didn't know that what they taught me would one day apply to the software industry. Israel "El" Gabriel, Richard Adama, Olga Maynard, Jillana, James Penrod, Donald Bradburn, James Jones, Paul Shipton, and Paul Sutherland.

The "choreographers and coaches" at the Software Engineering Institute who didn't know that I was in the wings learning from them. Watts Humphrey, Stan Rifkin, Priscilla Fowler, Mark Paulk, Mike Phillips, Bette Diemel, Chuck Meyers, Suzanne Garcia, Jim Over, Tim Olson, and Roselyn Whitney.

The SEPG members, managers, and engineers whom I have had the privilege of working with personally, a cast too numerous to name here, but who participated in developing, reviewing, and implementing what I have written about in this book. This book would not be possible without their efforts.

Paul Goodwin and Rich Liechty who were my mentors, and Nancy Lynd and Alan Jones who were my peers in the early days on the first SEPG at Unisys Corporation, learning and discovering how to apply the CMM even before the CMM was documented. Laurie Patton and Vin Ludwig who were my teammates on the next cycle of improvement, whose

support was instrumental in developing and implementing many of the process documents and files in this book. I am grateful for their support then and now.

Linda Lindquist who encouraged me to bring this work forward to be shared and leveraged with other SEPGs within the company. This led to the possibility of writing a book to further share and leverage these ideas. Linda is now at Xerox Corporation, but she worked closely with me during the early days, and I am grateful for her guidance and support during the early development of this book.

Jay Stockett, Larry Powers, Lance Callaghan, Paul Kraska, Gail Bertossi, Dave Parker, Frances DePonio, Dick Bland, and Joe Ringland, who were SEPG members at other sites in the company who were most actively involved in the sharing and leveraging, developing working relationships, and transferring knowledge and practice between SEPGs. I appreciate their enthusiasm and encouragement from the outside looking in, and from the inside looking out.

Lee Osterweil who founded the Southern California Software Process Improvement Network (SC SPIN) and to Debra Brodbeck, Leitha Purcell, Rick Hefner, Leia White, Kent Palmer, A. Windsor Brown, and Warren Scheinin, who continued to grow the SC SPIN after Lee's departure. (Lee planted the SC SPIN as a tree he never intended to sit under, and the others successfully took care of tending the tree.) This group provided me with access to other people in the software process improvement industry whose ideas stimulated or affirmed my work. Later the SC SPIN provided me with my first outside audience, giving me the encouragement and confidence to continue, and to reach further.

I would also like to acknowledge the contributions of three external reviewers who contributed much to help me improve my draft manuscripts: Michael Sturgeon of the Software Technology Support Center and Transitions Management, Inc.; Marek Wakulczyk of the Canadian Air Force and Small Bear Management Services; and Stan Rifkin of Master Systems, Inc., co-author of the SEI Technical Report, *Software Engineering Process Group Guide*, CMU/SEI-90-TR-24.

Michael Sturgeon for encouraging me to submit proposals to international conferences, such as the SEPG Conference and the Software Technology Conference, and attending and celebrating each successful presentation. Mike helped me develop my vision for this book, a vision that became much bigger than I first imagined. I am grateful for his understanding and support during the development of this book, especially for not letting me get stuck with my worst case of writer's block, and for always believing that I would make it to the finish line of this writer's marathon. Mike continuously encourages me to grow.

Marek Wakulczyk for his witty review comments and his encouragement and guidance that kept me thinking about how to keep the reader interested, and how to get more movement into the words. Marek helped me bring my draft manuscripts more into alignment with my vision, providing me with the outside perspective that I couldn't get when I was too close to the work. Because of the similarities in our approaches, Marek continues to be a remarkable mirror for me.

Stan Rifkin for his subtle yet powerful influence on my performance. Before I met him, his writing helped my SEPG to do the work. After I met him, his review comments helped me shape this book. Stan was my toughest critic when I was off balance, and my

greatest fan when I struck the perfect balance. In my early drafts, I had the right idea but the wrong tone of voice in my writing style. Stan helped me find my voice, and then he encouraged me to use it, reassuring me to have confidence that others would hear it. Stan continuously builds my confidence in my own performance.

Carole Hollinger who supported me as my manager for a majority of the months that I spent writing the manuscript. Carole encouraged me when I got my case of writer's block, reminding me of the importance of fresh ideas, and telling me to stop worrying about what others will think and to trust that the readers will find the ideas they need.

Chris Harding who supported me as my manager as I completed this book. Though many people start to write a book, fewer finish it. Without his help, I might have remained one of the former instead of the latter, remaining incomplete instead of complete. I am grateful for his encouragement in helping me make the transition to a new beginning as a published author.

Special Thanks to my "supporting cast" from the Unisys Authors Sponsorship Program and Addison Wesley Longman, especially Peter Gordon, Dot Malson, Ellen Gwynn, Carla Freeman, Jim Senior, Helen Goldstein, Jacquelyn Young, and the anonymous reviewers.

Extra special thanks to my "supporting cast" at home, my husband Christopher J. Caputo, my children Curtis Caputo and Vanessa Caputo, my parents Bailey Daugherty and Evelyn Daugherty, and my other parents Ralph Caputo and Carol Caputo. Without their time, patience, and understanding, this performance would not have happened. I am especially grateful to my husband Christopher J. Caputo for reminding me that this book is just another performance, and that I have a life before, during, and after the performance. Thanks to him, I do.

郑重声明

责任编辑　康兆华

封面设计　张　楠

责任印制　陈伟光

Contents

Foreword vii

Preface ix

Acknowledgments xiii

1 Introduction: Performing Software Process Improvement 1
- Practical Experience and Theoretical Models 1
- The SEPG Function 2
- Choreography as a Metaphor for Software Process Improvement 3
- Improvement Infrastructure 8
- Envisioning, Encoding, and Enacting 11

2 Assumptions: Turning the Culture Around 19
- Creating Movement 19
- Communications, Expectations, and Assumptions 20
- SEPG Activities: Visible and Invisible 22
- Assumptions in the Capability Maturity Model 24

3 Assessments: Looking at Your Reflection in the Mirror 33
- Frame of Reference 33
- Capability Maturity as Organizational Self-Awareness 35
- Assessment Techniques 41

4 Improvement Cycles: Dancing With the Rhythms 49
- Finding the Steps: Implementation Life Cycles 49
- Finding the Music: The Stages of Transition 51
- Finding the Tempo: The Pulse of Change 53
- Confidence 58
- Using Feedback to Manage Change 63

5 Action Plans: Preparing for Movement 65
- Preparation, Action, Finish 65
- Visualizing the Finishing Position 69
- Sequence of Steps 70
- Moving from One Step to the Next 74
- Creating a New Focus 74
- Making Progress Being Practical 78

6 Process Documents: Collaborating to Define the Steps 81
- Collaboration 81
- Starting from Scratch 82
- Starting with Baggage 90
- Process Definition Pitfalls: Shoulds and Passive Voices 93
- People and Communication, Data and Information (PCDI) 97

7 Process Implementation: Lifting the Spirits of the Performers 101
- Different Strokes for Different Folks 103
- Measuring Overall Progress 107
- Adoption Techniques 110
- From Adoption to Institutionalization 115
- Postmortem and Risk Analysis Techniques 116

8 Perspectives: Dancing Together and Creating a Better Performance 125
- Partnership and Teamwork 125
- Levels of Shared Knowledge and Awareness 127
- Three-Dimensional Perspective 130
- Development and Discovery 131
- Discovering Level 5 Within Each Level of the CMM 134
- Encouraging Words 138

Appendices

A Assumptions Worksheet 141
- Worksheets 142

B CMM Overview Workshop 145
- Presentation Materials and Worksheets 146

C Project Manager Interviews 167
- Questionnaire and Results Format 168

D Great Performance Worksheet 179
- Worksheet 181

E CMM Key Process Area Checklist 183
- Worksheet 184

F Simple Action Plan 187
- Worksheet 188

G Transition Preparation Method Charts 189
- Charts 191

H Results, Needs, Activities Worksheet 193
- Worksheet 194

I Is This Project Worthwhile? Worksheet 195
- Questionnaire 196

J Sample Processes for Level 2 197
- Project Management Policy 198
- Project Management Processes Overview 200
- Requirements Management Process 212
- Project Planning Process 227
- Estimation Process 235
- Commitment Control Process 242
- Project Tracking Process 247
- Project Management Review Process 253
- Software Quality Assurance Process 262

K Project Notebook Table of Contents 267
- Template Samples 268

L Process Definition Templates 271
- Template Samples 272

M CMM Action Planning Workshop Technique 277
- Presentation Materials and Worksheet 278