

Reference Point

Software Configuration Management Resources

by Steve Berczuk

IN THIS ISSUE, STEVE BERCUK SHARES USEFUL RESOURCES FOR SOFTWARE CONFIGURATION Management. Steve has been an active member of the software patterns community since the first PLoP (Pattern Languages of Programs) conference in 1993, and he did early work on the relationship between organization, software architecture, and design patterns. He co-authored (with Brad Appleton) the book *Software Configuration Management Patterns: Effective Teamwork, Practical Integration*.

Software configuration management (SCM) includes such practices as configuration identification, configuration control, status accounting, artifact reviews, build management, process management, and teamwork. These practices define how an organization builds and releases products, and how it identifies and tracks changes. For most people involved in developing and testing software, SCM practices manifest themselves in version control tools and how their organizations use those tools.

New SCM practices are met with either fear or anticipation. Good SCM practices add a measure of reproducibility, control, and security to the work environment. However, sometimes even good practices can seem restrictive when they are not well motivated by the needs of the project. And people can inflict too much SCM process on the organization, proving again that you can have too much of a good thing. If a practice is too rigid, or just too different, team members often complain that SCM gets in the way, because change tracking and integration—two essential aspects to building a reliable product—are harder to do than they should be.

It is essential that team members embrace SCM practices because some coordination is necessary to having a reliable, productive product development environment. To ensure acceptance, first determine which practices are appropriate, and then choose the right tool based on your needs and budget. At a minimum you will want to use a version control tool to track changes to some (or all) of your development artifacts, and also make the system easy to use. The following references will help you understand how to establish SCM practices that are appropriate for your organization.

The Basics

***Software Release Methodology*, Michael Bays**

ISBN 0136365647, Prentice Hall, 1999, \$50.00, 247 pp.

This is an excellent overview of the basic concepts of SCM.

***Rapid Development: Taming Wild Software Schedules and Code Complete: A Practical Handbook of Software Construction*, Steve McConnell**

ISBN 1556159005, Microsoft Press, 1996, \$24.50, 647 pp.

ISBN 1556154844, Microsoft Press, 1993, \$24.50, 857 pp.

Both of these books describe how important the activities at the heart of SCM, including change management and consistent build practices, are to effective coding.

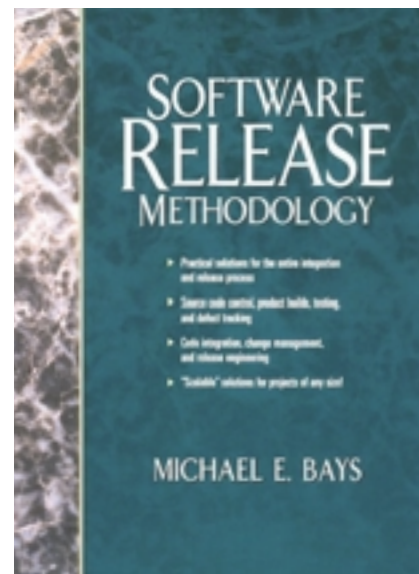
***Software Configuration Management Patterns: Effective Teamwork, Practical Integration*, Steve Berczuk with Brad Appleton**

ISBN 0201741172, Addison-Wesley, 2002, \$39.99, 256 pp.

This book describes how some proven SCM practices fit together in a way that helps teams work together without adding too much overhead. The related Web site www.berczuk.com/scm-patterns-book has pointers to other useful resources.

***Are Your Lights On? How to Figure Out What the Problem Really Is*, Donald C. Gause and Gerald M. Weinberg**

ISBN 0932633161, Dorset House, 1990, \$13.95, 176 pp.



An excellent overview of basic SCM concepts.

Reference Point

One problem people encounter when they use SCM tools and techniques is that they do “too much,” and their system ends up hindering rather than helping them. This book is not about SCM but rather how to figure out what is really wrong in a system. It is an entertaining read that teaches an important and often undervalued skill.

Tools

Open Source Development with CVS, Karl Fogel and Moshe Bar

ISBN 1932111204, Paraglyph Publishing, 2002, \$39.99, 368 pp.

This book describes how to use CVS, a very common open source version control system. Along the way, it explains what the basic version control concepts are. The sections on open source development illustrate how team structure factors into the mix.

The CVS Web site

www.cvshome.org

The Concurrent Versions System (CVS) Web site contains downloads for CVS and related tools, including CVS documentation.

Ant

<http://jakarta.apache.org/ant/>

Ant is a very versatile tool for automating SCM processes, including builds, deployments, and releases.

Software Configuration Management Strategies and Rational ClearCase: A Practical Introduction, Brian White and Geoffrey Clemm

ISBN 0201604787, Addison-Wesley, 2000, \$39.95, 336 pp.

This book describes the concepts in ClearCase and how to use them. The early chapters have a good description of general SCM concepts.

General Resources

UCM Central

www.snuffbear.com/ucmcentral.htm

This site provides straightforward explanations of CM in a light-hearted manner.

The Configuration Management Yellow Pages

www.cmtoday.com/yp/configuration_management.html

As the name implies, this site is an index of resources for tools, jobs, and other information important to SCM practitioners.

CM Crossroads—Online Community and Resource Center for Configuration Management

www.cmcrossroads.com

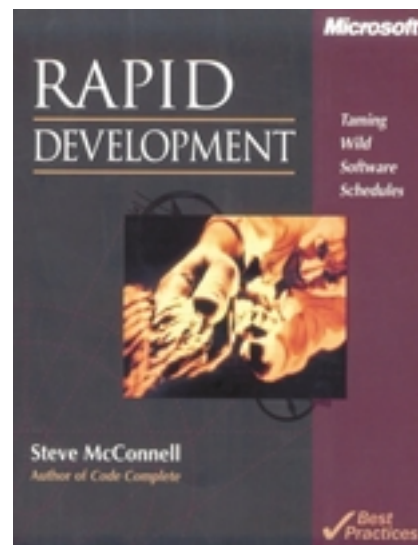
This Web site provides discussion forums, resources, tools, and news.

ACME—Assembling Configuration Management Environments (for Software Development)

www.acme.bradapp.net

You will find a comprehensive listing of SCM definitions and best practices located on this site. If you want to explore some of the points in this article in more detail, this site is a good place to start. **STQE**

Steve Berczuk (steve@berczuk.com) has been developing object-oriented software applications since 1989. You can find out more about this topic and others at his Web site: www.berczuk.com.



This book conveys the importance of SCM activities for effective coding.

STQE magazine is produced by Software Quality Engineering.