Sustainable Software Development: An Agile Perspective

Kevin Tate, Addison-Wesley Professional, 2005, \$39.99,
ISBN: 0321286081

Our software engineering community has for decades flirted with the idea of applying the rigor of other engineering disciplines to the development of software. This book boldly argues against this metaphor. Buildings are expensive to modify and typically static, whereas software

is cheap to modify and evolves over its lifetime.

Instead, author Kevin Tate argues that an appropriate metaphor is a coral reef: an ecosystem of developers, customers, suppliers, distributors, and competitors that live on top of the software, in the same way that a reef's organisms live around the coral. Both the coral and the software evolve with their surrounding ecosystems.

The book distinguishes itself from other agile programming books by taking a wider view of the field, covering not only the project management side of agile practices, but also developer collaboration and technical excellence. It starts by arguing that the goal of sustainability comes into play by recognizing that a project's progress depends on the competition between negative stresses (user requirements, disruptive technologies and business models, external dependencies, competition, and cost management) and positive controls (collaboration, methodology, expertise, decision making, leadership, culture, and simplicity). When the negative stresses outweigh the counteraction of a project's controls, the project enters into a death spiral of diminishing productivity.

The remainder of the book is organized around a chapter for each of the four principles that should guide sustainable development: defect prevention, a working product, emphasis on design, and continual refinement. With another apt metaphor, Tate advises developers to juggle the four principles of sustainable development while working on product features and fixing bugs.

The text is full of interesting ideas and illuminating sidebars discussing real-world cases, but as a result the reader can occasionally get lost among them, losing focus on the argument and the course of thought. Nevertheless, this is a book that both developers and managers will appreciate and value. Its advice is important, understand-

able, and practical: a gift to the software engineering community.

—D. Spinellis

Hacking Exposed: Web Applications, 2nd edition

Joel Scambray, Mike Shema, Caleb Sima, McGraw-Hill
Osborne Media, 2006, \$49.99, ISBN: 0072262990

Many years ago, the "Hacking Exposed" book series started covering security from a hacker's perspective. Since the security landscape has become more complex, the series now covers the multiple facets of network and system security, and includes books on specific systems (Linux, Windows, Cisco), as well as wireless networks (forthcoming in 2007).

This book is dedicated to the security of Web applications and associated service deployment architectures. It is written from an attacker's point of view and follows the basic steps that an attacker takes. It starts with a description of the reconnaissance phase (fingerprinting the application and the supporting Web server) and moves on to more intrusive attacks, roughly divided into those against authentication methods, bypassing authorization mechanisms, abusing the input-validation procedures, and stealing sensitive information. These subjects are presented well, both from a conceptual point of view and through examples drawn from real-world cases. There is also a relatively short chapter addressing the security and vulnerabilities of Web services.

Hacking on the server side is only one viewpoint of Web security. The typical end user may also get hacked just by visiting malicious or compromised Web sites. One chapter in the book reviews the most famous exploits and vulnerabilities related to these issues.

This book meets high expectations. It is fun and easy to read. It covers in sufficient depth the technical details and underlying system and software-specific issues. The technical level is somewhere between intermediate and advanced, thus appealing to a broad range of readers. Webmasters will learn how to check their servers for the most common security flaws, programmers will appreciate the contents on securing their code, and typical readers will get a comprehensive picture of the status of Web security today. I recommend this truly exceptional book to all of these readers.

—Radu State

Reprinted from *Computing Reviews*, © 2006 ACM, <http://www.reviews.com>