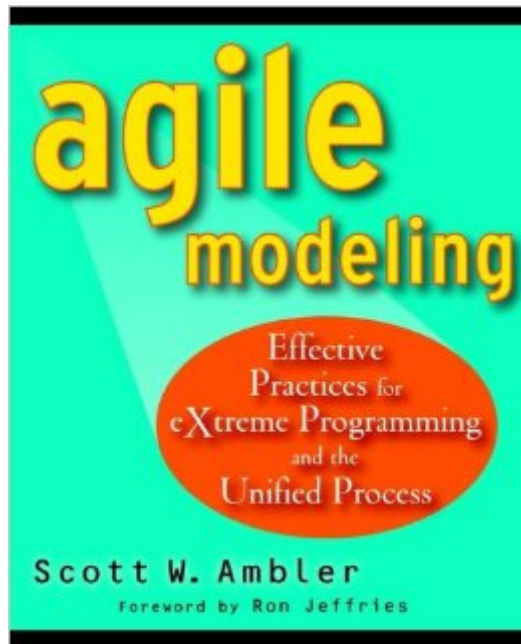## The book was found

Agile Modeling: Effective Practices For EXtreme Programming And The Unified Process

Synopsis

The first book to cover Agile Modeling, a new modeling technique created specifically for XP projects eXtreme Programming (XP) has created a buzz in the software development community-much like Design Patterns did several years ago. Although XP presents a methodology for faster software development, many developers find that XP does not allow for modeling time, which is critical to ensure that a project meets its proposed requirements. They have also found that standard modeling techniques that use the Unified Modeling Language (UML) often do not work with this methodology. In this innovative book, Software Development columnist Scott Ambler presents Agile Modeling (AM)-a technique that he created for modeling XP projects using pieces of the UML and Rational's Unified Process (RUP). Ambler clearly explains AM, and shows readers how to incorporate AM, UML, and RUP into their development projects with the help of numerous case studies integrated throughout the book. AM was created by the author for modeling XP projects-an element lacking in the original XP design The XP community and its creator have embraced AM, which should give this book strong market acceptance Companion Web site at www.agilemodeling.com features updates, links to XP and AM resources, and ongoing case studies about agile modeling.

Book Information

File Size: 6616 KB

Print Length: 400 pages

Publisher: Wiley; 1 edition (April 21, 2008)

Publication Date: April 21, 2008

Language: English

ASIN: B000SELMZK

Text-to-Speech: Enabled

X-Ray: Not Enabled

Word Wise: Not Enabled

Lending: Not Enabled

Enhanced Typesetting: Not Enabled

Best Sellers Rank: #1,105,002 Paid in Kindle Store (See Top 100 Paid in Kindle Store)   #50 inÂ Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Quality Control   #235 inÂ Books > Business & Money > Management & Leadership > Quality Control & Management > Agile   #1541 inÂ Kindle Store > Kindle eBooks > Computers &

Customer Reviews

An adept application of common sense--and the author's significant experience--to the use of models in software development.A model can be almost anything that developers make to describe the software that they build--just like an architect's drawings.A given software development effort might call for any number of different types of models including data models, class models, sequence diagrams, dataflow diagrams, statechart diagrams, etc. The set of models used on any particular project will depend partly on the nature of the project and partly on the preferred methodology of the software developers.Agile Modeling (AM) is not itself a software development methodology. It is a collection of principles and practices to follow when using models to develop software according to a methodology like Rational Unified Process (RUP) or eXtreme Programming (XP). Many of the practices derive from an application of XP concepts.AM challenges a number of practices widely followed (or at least preached) in organizations developing software: 1. Specializing personnel in producing a single type of model 2. Dedicating work sessions to producing a single type of model 3. Saving models after the software is developed 4. Keeping models up-to-date during and after the development project 5. Using sophisticated software to assist in modeling 6. Finishing models before coding softwareAM does not in all cases prohibit these practices, but it emphasizes that the purpose of a software development project is to develop software--not just to develop models. The practices of AM help to keep models in their proper subordinate relation to the working software that is the true goal of any development project.

The hype that grew around eXtreme Programming (XP) in the year 2001, and the publication of now almost 2 dozen books devoted to XP has not cleared up the original vagueness of what practices are allowed and what aren't. To a casual observer the XP culture seems replete with "Thou shalt not's"Ã Â don't do Big Requirements Up Front, don't do Big Design Up Front, don't build models because it's only the code that matters, etc. For those of us who think it's important to have a map of where you are going before you start a long trip, some of the radicalness of XP wasÃ Â well, too radical and too unplanned.Scott Ambler's new book, Agile Modeling, addresses a sane middle ground between the apparent unstructured XP and the overly structured approach in the Rational Unified Process (RUP). Agile Modeling is arguably Ambler's best book to date. It conveys an approach that is truly a confluence of best practices and does so in a very readable, accessible presentation.Ambler presents Agile Modeling (AM) as a set of values, principles and practices. AM's

values are borrowed directly from those in XP: Communication, Simplicity, Feedback and Courage, with Ambler's addition of "humility". In my experience this additional value is a defining characteristic of an effective modeler and mentor. From these 5 values Ambler defines the principles for AM, including: Software is your Primary Goal, Travel Light, Embrace Change, use Multiple Models, etc.It is interesting to me that no one would refute these principles: they are too close to Motherhood and Apple pie. But it is disturbing how seldom I see any of these principles actually embraced by the dozen or more organizations I provide mentoring services to each year.

This is a mix of good, bad, and annoyingGood: the author really does know a lot about modeling (except data modeling, see "Bad") and gives good explanations and examples of many aspects of modeling at many stages in the development process. If you can plough through his 350+ pages, you will have found many stimulating and practical concepts and some good advice on implementing them.A very good chapter is Chapter 29 - a discussion of how to implement Agile Modeling - or really, any agile practice - in a usually hostile world. Some battle scars showing here!I also like that he does not consider the UML the be-all and end-all of modeling tools. Like him, I've found good use for the trusty old DFD (Data Flow Diagram) of the 70's, where appropriate.And his overall message - that the agile approach can extend to your design and modeling task, not just code, and the implications for minimizing the documentation effort - is very strong.I find his reference to quick diagrams "on the back of a napkin" a bit overdone. Sure, the quick informal diagram is excellent, but paper napkins are not the best medium! Hand-drawn on a piece of paper, or a card, sure...if you are discussing models in a bar or restaurant with that degree of focus...get a life!Whiteboard and digital camera can certainly be used much more than they are. But the overall point is excellent: that when you are documenting (and he has some difficulty separating out "modeling" from "documenting" and acknowledges the problem) you are not creating the end-product, and there is a cost for that. "Travel light" - yes. As Einstein said "Everything should be as simple as possible, but no simpler."Bad: his data model example is terrible.

Agile Modeling: Effective Practices for eXtreme Programming and the Unified Process Programming #8:C Programming Success in a Day & Android Programming In a Day! (C Programming, C++programming, C++ programming language, Android , Android Programming, Android Games) Programming #57: C++ Programming Professional Made Easy & Android Programming in a Day (C++ Programming, C++ Language, C++for beginners, C++, Programming ... Programming, Android, C, C Programming) Agile in a Flash: Speed-Learning Agile Software Development

(Pragmatic Programmers) Programming #45: Python Programming Professional Made Easy & Android Programming In a Day! (Python Programming, Python Language, Python for beginners, ... Programming Languages, Android Programming) DOS: Programming Success in a Day: Beginners guide to fast, easy and efficient learning of DOS programming (DOS, ADA, Programming, DOS Programming, ADA ... LINUX, RPG, ADA Programming, Android, JAVA) ASP.NET: Programming success in a day: Beginners guide to fast, easy and efficient learning of ASP.NET programming (ASP.NET, ASP.NET Programming, ASP.NET ... ADA, Web Programming, Programming) C#: Programming Success in a Day: Beginners guide to fast, easy and efficient learning of C# programming (C#, C# Programming, C++ Programming, C++, C, C Programming, C# Language, C# Guide, C# Coding) FORTRAN Programming success in a day:Beginners guide to fast, easy and efficient learning of FORTRAN programming (Fortran, Css, C++, C, C programming, ... Programming, MYSQL, SQL Programming) Prolog Programming; Success in a Day: Beginners Guide to Fast, Easy and Efficient Learning of Prolog Programming (Prolog, Prolog Programming, Prolog Logic, ... Programming, Programming Code, Java) Parallel Programming: Success in a Day: Beginners' Guide to Fast, Easy, and Efficient Learning of Parallel Programming (Parallel Programming, Programming, ... C++ Programming, Multiprocessor, MPI) Programming: Computer Programming for Beginners: Learn the Basics of Java, SQL & C++ - 3. Edition (Coding, C Programming, Java Programming, SQL Programming, JavaScript, Python, PHP) Raspberry Pi 2: Raspberry Pi 2 Programming Made Easy (Raspberry Pi, Android Programming, Programming, Linux, Unix, C Programming, C+ Programming) Android: Programming in a Day! The Power Guide for Beginners In Android App Programming (Android, Android Programming, App Development, Android App Development, ... App Programming, Rails, Ruby Programming) R Programming: Learn R Programming In A DAY! - The Ultimate Crash Course to Learning the Basics of R Programming Language In No Time (R, R Programming, ... Course, R Programming Development Book 1) Agile Software Development, Principles, Patterns, and Practices Agile Principles, Patterns, and Practices in C# Business Process Technology: A Unified View on Business Processes, Workflows and Enterprise Applications Extreme Coloring Amazing World: Relax and Unwind, One Splash of Color at a Time (Extreme Art!) Extreme Love (Love to the Extreme Book 1)