The Elements Of Uml 2 0 Style

This book constitutes a collection of the best papers selected from the 12 workshops and 3 tutorials held in conjunction with MODELS 2008, the 11th International Conference on Model Driven Engineering Languages and Systems, in Toulouse, France, September 28 - October 3, 2008. The contributions are organized within the volume according to the workshops at which they were presented: Model Based Architecting and Construction of Embedded Systems (ACES-MB); Challenges in Model Driven Software Engineering (CHAMDE); Empirical Studies of Model Driven Engineering (ESMDA); Models@runtime; Model Co-evolution and Consistency Management (MCCM); Model-Driven Web Engineering (MDWE); Modeling Security (MODSEC); Model-Based Design of Trustworthy Health Information Systems (MOTHIS); Non-functional System Properties in Domain Specific Modeling Languages (NFPin DSML); OCL Tools: From Implementation to Evaluation and Comparison (OCL); Quality in Modeling (QIM); and Transforming and Weaving Ontologies and Model Driven Engineering (TWOMDE). Each section includes a summary of the workshop. The last three sections contain selected papers from the Doctoral Symposium, the Educational Symposium and the Research Project Symposium, respectively.
The popular Unified Modeling Language (UML) is both a language and notation developed by the Object Management Group (OMG) used to design and create specifications for software systems. With the recent release of version 2.0 UML, the OMG has started the OMG-Certified UML Professional Program to provide an objective measure of UML knowledge. As a certified UML professional a developer has an important credential to present to employers and clients. Certification also benefits companies looking for skilled UML practitioners by giving them a basis for making hiring and promotion decisions. UML 2 Certification Guide is the only official study guide to passing the new UML exams. This book systematically covers all of the topics covered in the exams, and has been carefully reviewed by the OMG. The book begins by assuming only a basic knowledge of UML and then progresses far enough to allow a reader to pass both the fundamental and the intermediate level exams. Along the way the book also covers topics that are not in introductory books on UML but that are necessary to pass the exams. Tim Weilkiens is considered one of the top ten experts on UML, and both authors have extensive experience training developers to successfully take the exams. The official certification resource Assumes a basic knowledge of UML so that you can focus immediately on the exams Written by two authors known for their skill as trainers, consultants, and developers Developed systematically to enable you to master all exam topics—without exception Covers the use of UML for applications, as required by the exams, both inside and outside of the realm of software development Includes a practice exam, glossary, list of books, and website information
Requirements engineering tasks have become increasingly complex. In order to ensure a high level of knowledge and competency among requirements engineers, the International Requirements Engineering Board (IREB) developed a standardized qualification called the Certified Professional for Requirements Engineering (CPRE). The certification defines the practical skills of a requirements engineer on various training levels. This book is designed for self-study and covers the curriculum for the Certified

Professional for Requirements Engineering Foundation Level exam as defined by the IREB. **The 2nd edition** has been thoroughly revised and is aligned with the curriculum Version 2.2 of the IREB. In addition, some minor corrections to the 1st edition have been included. **About IREB:** The mission of the IREB is to contribute to the standardization of further education in the fields of business analysis and requirements engineering by providing syllabi and examinations, thereby achieving a higher level of applied requirements engineering. The IRE Board is comprised of a balanced mix of independent, internationally recognized experts in the fields of economy, consulting, research, and science. The IREB is a non-profit corporation. For more information visit www.certified-re.com

This book constitutes the proceedings of the 4th EuroSymposium on Systems Analysis and Design, SIGSAND/PLAIS 2011, held in Gda?sk, Poland, in September 2011. The objective of this symposium is to promote and develop high-quality research on all issues related to systems analysis and design (SAND). It provides a forum for SAND researchers and practitioners in Europe and beyond to interact, collaborate, and develop their field. The 9 papers were carefully reviewed and selected from 20 submissions. An additional revision took place after the conference to incorporate discussion results from the presentation. The contributions are organized into topical sections on business process modeling, integrated systems development, and software development.

The only study to integrate a platform-driven decision framework with the model-driven architecture (MDA), this in-depth, practical guide explains how to combine three technology areas--MDA, ontologies, and software product lines--in order to integrate several platform-specific software products into a single MDA. Further exploring concepts such as knowledge-representation systems, artificial intelligence, and the "semantic web," this complete resource reveals why these technologies are so highly relevant today and speculates on how they will impact future software development.

Thomsen and Hansen give easy-to-understand examples and provide readers with everything they need to create Enterprise solutions with .NET.

"This book displays how to effectively map and respond to the real-world challenges and purposes which software must solve, covering domains such as mechatronic, embedded and high risk systems, where failure could cost human lives"--Provided by publisher.

This book constitutes the refereed proceedings of the 8th International Conference on Data Warehousing and Knowledge Discovery, DaWak 2007, held in Regensburg, Germany, September 2007. Coverage includes ETL processing, multidimensional design, OLAP and multidimensional model, cubes processing, data warehouse applications, frequent itemsets, ontology-based mining, clustering, association rules, miscellaneous applications, and classification.

Gain the skills to effectively plan software applications andsystems using the latest version of UML UML 2 represents a significant update to the UML specification,from providing more robust mechanisms for modeling workflow andactions to making the modeling language more executable. Now in itssecond edition, this bestselling book provides you with all thetools you'll need for effective modeling with UML 2. The authorsget you up to speed by presenting an overview of

UML and its mainfeatures. You'll then learn how to apply UML to produce effectivediagrams as you progress through more advanced topics such asuse-case diagrams, classes and their relationships, dynamicdiagrams, system architecture, and extending UML. The authors takeyou through the process of modeling with UML so that you cansuccessfully deliver a software product or information managementsystem. With the help of numerous examples and an extensive case study,this book teaches you how to: * Organize, describe, assess, test, and realize use cases * Gain substantial information about a system by usingclasses * Utilize activity diagrams, state machines, and interactiondiagrams to handle common issues * Extend UML features for specific environment or domains * Use UML as part of a Model Driven Architecture initiative * Apply an effective process for using UML The CD-ROM contains all of the UML models and Java?TM code for acomplete application, Java?TM 2 Platform, Standard Edition, Version1.4.1, and links to the Web sites for vendors of UML 2 tools.

This book constitutes the refereed proceedings of the Third International Conference on Graph Transformations, ICGT 2006. The book presents 28 revised full papers together with 3 invited lectures. All current aspects in graph drawing are addressed including graph theory and graph algorithms, theoretic and semantic aspects, modeling, tool issues and more. Also includes accounts of a tutorial on foundations and applications of graph transformations, and of ICGT Conference satellite events.

This book constitutes the refereed proceedings of the 7th European Conference on Modelling Foundations and Applications, held in Birmingham, UK, in June 2011. The 19 revised full foundations track papers and 5 revised full applications track papers presented were carefully reviewed and selected from 61 submissions; also included are 5 workshop summaries and abstracts of 4 tutorials. The papers are organized in topical sections on model execution, model analysis, methodology, model management, model transformation, variability analysis and ADLs, and domain-specific modeling.

UML stands for Unified Modeling Language used for creating object-oriented, meaningful documentation models for any software system present. It provides us a way to develop rich models that describe the working of any software/hardware systems. UML serves a great way of creating professional documentation which is a necessary part of any project development. Here is what is covered in the book – Chapter 1: UML Diagrams: Versions, Types, History, Tools, Examples 1.What is UML? 2.Why use UML? Complete History 3.UML Versions 4.Characteristics of UML 5.Conceptual model 6.UML Diagrams 7.UML Tools Chapter 2: UML Notation Tutorial: Symbol with Examples 1.What is a model? 2.UML Building Blocks 3.Things 4.Relationships 5.Diagrams Chapter 3: UML Relationships with EXAMPLE: Dependency, Generalization, Realization 1.Association 2.Dependency 3.Generalization 4.Realization 5.Composition

6.Aggregation Chapter 4: UML Association vs Aggregation vs Composition with EXAMPLE 1.Association 2.Composition 3.Aggregation 4.Association vs. Aggregation vs. Composition Chapter 5: UML Class Diagram Tutorial with Examples 1.What is Class? 2.What is Class Diagram? 3.Benefits of Class Diagram 4.Essential elements of A UML class diagram 5.Aggregation vs. Composition 6.Abstract Classes 7.Example of UML Class Diagram 8.Class Diagram in Software Development Lifecycle 9.Best practices of Designing of the Class Diagram Chapter 6: What is UML Object Diagram? Tutorial with Example 1.What is a Class Diagram? 2.What is an Object Diagram? 3.How to draw an object diagram? 4.Purpose of an object diagram: 5.Applications of Object Diagrams: 6.Class vs. Object Diagrams Chapter 7: UML Use Case Diagram: Tutorial with EXAMPLE 1.What is the Use Case Diagram? 2.Why Use-Case diagram? 3.Use-case diagram notations 4.How to draw a use-case diagram? 5.Tips for drawing a use-case diagram 6.An example of a use-case diagram 7.When to use a use-case diagram? Chapter 8: State Machine Diagram: UML Tutorial with EXAMPLE 1.What is a State Machine Diagram? 2.Why State Machine Diagram? 3.Notation and Symbol for State Machine 4.Types of State 5.How to draw a Statechart diagram? 6.When to use State Diagrams? 7.Example of State Machine 8.State machine vs. Flowchart Chapter 9: UML Activity Diagram: What is, Components, Symbol, EXAMPLE 1.What is an Activity Diagram? 2.Components of Activity Diagram 3.Why use Activity Diagrams? 4.Activity Diagram Notations 5.How to draw an activity diagram? 6.Example of Activity Diagram 7.When Use Activity Diagram Chapter 10: Interaction, Collaboration, Sequence Diagrams with EXAMPLES 1.What is Interaction diagram? 2.Purpose of an Interaction Diagram 3.Important terminology 4.Types of Interaction diagram and Notations 5.Sequence Diagram 6.What is the Collaboration diagram? 7.Timing diagram 8.How to draw a Interaction diagram? 9.Use of an interaction diagram Chapter 11: Component Diagram: UML Tutorial with EXAMPLE 1.What is Component Diagram? 2.Component diagram Notations 3.What is a Component? 4.Why use Component Diagram? 5.When to use Component Diagram? 6.How to draw a component diagram 7.Example of a component diagram Chapter 12: Deployment Diagram: UML Tutorial with EXAMPLE 1.What is Deployment Diagram? 2.Purpose of a deployment diagram 3.Deployment Diagram Symbol and notations 4.What is an artifact? 5.What is a node? 6.How to draw a deployment diagram? 7.Example of a Deployment diagram 8.When to use a deployment diagram? Click the BUY button now and download the book now to start learning UML. Learn it fast and learn it well. Pick up your copy today by clicking the BUY NOW button at the top of this page!

Uses friendly, easy-to-understand For Dummies style to helpreaders learn to model systems with the latest version of UML, themodeling language used by companies throughout the world to developblueprints for complex computer systems Guides programmers, architects, and business analysts throughapplying UML to design large, complex enterprise applications thatenable scalability, security, and robust execution Illustrates concepts with mini-cases from

different businessdomains and provides practical advice and examples Covers critical topics for users of UML, including objectmodeling, case modeling, advanced dynamic and functional modeling,and component and deployment modeling A detailed and practical book and eBook walk-through showing how to apply UML to real world development projects This book presents the technical program of the International Embedded Systems Symposium (IESS) 2009. Timely topics, techniques and trends in embedded system design are covered by the chapters in this volume, including modelling, simulation, verification, test, scheduling, platforms and processors. Particular emphasis is paid to automotive systems and wireless sensor networks. Sets of actual case studies in the area of embedded system design are also included. Over recent years, embedded systems have gained an enormous amount of proce- ing power and functionality and now enter numerous application areas, due to the fact that many of the formerly external components can now be integrated into a single System-on-Chip. This tendency has resulted in a dramatic reduction in the size and cost of embedded systems. As a unique technology, the design of embedded systems is an essential element of many innovations. Embedded systems meet their performance goals, including real-time constraints, through a combination of special-purpose hardware and software components tailored to the system requirements. Both the development of new features and the reuse of existing intellectual property components are essential to keeping up with ever more demanding customer requirements. Furthermore, design complexities are steadily growing with an increasing number of components that have to cooperate properly. Embedded system designers have to cope with multiple goals and constraints simul- neously, including timing, power, reliability, dependability, maintenance, packaging and, last but not least, price. Globe-trotting travelers have long resorted to handy, pocket-size dictionaries as an aid to communicating across the language barrier. Dan Pilone's UML 2.0 Pocket Reference is just such an aid for on-the-go developers who need to converse in the Unified Modeling Language (UML). Use this book to decipher the many UML diagrams you'll encounter on the path to delivering a modern software system. Updated to cover the very latest in UML, you'll find coverage of the following UML 2.0 diagram types: Class diagrams Component diagrams* Sequence diagrams* Communication diagrams* Timing diagrams* Interaction Overview diagrams* Package diagrams* Deployment diagrams* Use case diagrams Composite structure diagrams* Activity diagrams* Statechart diagrams* * New or expanded coverage in this edition Also new in this edition is coverage of UML's Object Constraint Language (OCL). Using OCL, you can specify more narrowly the functionality described in a given diagram by recording limits that are the result of business rules and other factors. The UML 2.0 Pocket Reference travels well to meetings and fits nicely into your laptop bag. It's near impossible to memorize all aspects of UML, and with this book along, you won't have to.

UML stands for Unified Modeling Language used for creating object-oriented, meaningful documentation models for any software system present. It provides us a way to develop rich models that describe the working of any software/hardware systems. UML serves a great way of creating professional documentation which is a necessary part of any project development. Here is what is covered in the book - Chapter 1: UML Diagrams: Versions, Types, History, Tools, Examples What is UML? Why use UML? UML Versions Characteristics of UML Conceptual model UML Diagrams UML Tools Chapter 2: UML Notation Tutorial: Symbol with Examples What is a model? UML Building Blocks Things Relationships Diagrams Chapter 3: UML Relationships with EXAMPLE: Dependency, Generalization, Realization Association Dependency Generalization Realization Composition Aggregation Chapter 4: UML Association vs Aggregation vs Composition with EXAMPLE Association Composition Aggregation Association vs. Aggregation vs. Composition Chapter 5: UML Class Diagram Tutorial with Examples What is Class? What is Class Diagram? Benefits of Class Diagram Essential elements of A UML class diagram Aggregation vs. Composition Abstract Classes Example of UML Class Diagram Chapter 6: What is UML Object Diagram? Tutorial with Example What is a Class Diagram? What is an Object Diagram? How to draw an object diagram? Purpose of an object diagram Applications of Object Diagrams Chapter 7: UML Use Case Diagram: Tutorial with EXAMPLE What is the Use Case Diagram? Why Use-Case diagram? Use-case diagram notations How to draw a use-case diagram? Tips for drawing a use-case diagram Chapter 8: State Machine Diagram: UML Tutorial with EXAMPLE What is a State Machine Diagram? Why State Machine Diagram? Notation and Symbol for State Machine Types of State How to draw a Statechart diagram? When to use State Diagrams? Chapter 9: UML Activity Diagram: What is, Components, Symbol, EXAMPLE What is an Activity Diagram? Components of Activity Diagram Why use Activity Diagrams? Activity Diagram Notations How to draw an activity diagram? Chapter 10: Interaction, Collaboration, Sequence Diagrams with EXAMPLES What is Interaction diagram? Purpose of an Interaction Diagram Important terminology Types of Interaction diagram and Notations Sequence Diagram What is the Collaboration diagram? Timing diagram Chapter 11: Component Diagram: UML Tutorial with EXAMPLE What is Component Diagram? Component diagram Notations What is a Component? Why use Component Diagram? When to use Component Diagram? Chapter 12: Deployment Diagram: UML Tutorial with EXAMPLE What is Deployment Diagram? Purpose of a deployment diagram Deployment Diagram Symbol and notations What is an artifact? What is a node? How to draw a deployment diagram? Click the BUY button now and download the book now to start learning UML. Learn it fast and learn it well. Pick up your copy today by clicking the BUY NOW button at the top of this page! A coherent and integrated account of the leading UML 2 semantics work and the practical applications of UML semantics development With contributions from leading experts in the field, the book begins with an introduction to UML and goes

on to offer in-depth and up-to-date coverage of: The role of semantics Considerations and rationale for a UML system model Definition of the UML system model UML descriptive semantics Axiomatic semantics of UML class diagrams The object constraint language Axiomatic semantics of state machines A coalgebraic semantic framework for reasoning about interaction designs Semantics of activity diagrams Verification of UML models State invariants Model transformation specification and verification Additionally, readers are provided with expert guidance on how to resolve semantic problems and a section on applications of UML semantics with model analysis. UML 2 Semantics and Applications is an ideal resource for researchers and tool-builders working in UML, among others. It is also an excellent textbook for postgraduate teaching and research.

Contains standards and guidelines for creating UML diagrams that are concise and easy to understand.

In the increasingly competitive corporate sector, businesses must examine their current practices to ensure business success. By examining their social, financial, and environmental risks, obligations, and opportunities, businesses can re-design their operations more effectively to ensure prosperity. Sustainable Business: Concepts, Methodologies, Tools, and Applications is a vital reference source that explores the best practices that promote business sustainability, including examining how economic, social, and environmental aspects are related to each other in the company's management and performance. Highlighting a range of topics such as lean manufacturing, sustainable business model innovation, and ethical consumerism, this multi-volume book is ideally designed for entrepreneurs, business executives, business professionals, managers, and academics seeking current research on sustainable business practices.

Market_Desc: · Programmers who wish to understand the work products of analysis and design· Designers who want a formal tool for design· Analysts who want to learn how to communicate more effectively with business and technical team members and clients· Project Leads And Managers who want to understand the tools available to facilitate quality communication and specification of software requirements· Maintenance Teams and Managers who wish to improve the overall quality and timeliness of their product support. Special Features: · Up-to-date coverage including both the 1.4 and 2.0 UML specifications. Focuses on executable UML meaning the UML diagrams are the code, rather than viewing the diagrams as a necessary evil to complete before coding· Covers testing of UML diagrams, rather than leaving all testing until the application is coded About The Book: The UML is an industry standard specification for modeling, visualizing, and documenting software projects. You can think of UML as the equivalent of blueprints in a manufacturing metaphor. By applying UML, developers can decrease the high-probability of failure that plagues large application development projects while improving quality. Extending the construction metaphor, the growing concept of executable UML is analogous to computer-aided-manufacturing settings where the blueprint for a product actually drives the machines

This book constitutes the thoroughly refereed joint postproceedings of the satellite activities held at the 7th International Conference on the Unified Modeling Language, UML 2004, in Lisbon, Portugal in October 2004 complementing the main conference track. The book presents reports on the 10 workshops held at UML and covers a broad range of topics around systems modelling; these reports are compiled by the respective workshop organizers. Furthermore 12 revised reviewed papers from the industry track are included as well as 11 short papers corresponding to selected poster/demo presentations and a summary on the UML tools exhibition.

This volume contains papers presented at the International Conference on Software Process (ICSP 2009) held in Vancouver, Canada, during May 16-17, 2009. ICSP 2009 was the third conference of the ICSP series, continuing the software process workshops from 25 years ago. The theme of ICSP 2009 was "Processes to Develop Trustworthy Software." Software development takes place in a dynamic context of frequently changing technologies and limited resources. Teams worldwide are under increasing pressure to deliver trustworthy software products more quickly and with higher levels of quality. At the same time, global competition is forcing software development organizations to cut costs by rationalizing processes, outsourcing part or all of their activities, re- ing existing software in new or modified applications and evolving existing systems to meet new needs, while still minimizing the risk of projects failing to deliver. To address these difficulties, new or modified processes are emerging including lean and agile methods, plan-based product line development, and increased integration with systems engineering processes. Papers present research and real-world experience in many areas of software and systems processes impacting trustworthy software including: new software devel- ment approaches; software quality; integrating software and business processes; CMMI and other process improvement initiatives; simulation and modeling of so- ware processes; techniques for software process representation and analysis; and process tools and metrics.

For all developers who create models using the Unified Modeling Language (UML) 2.x The Elements of UMLTM 2.0 Style sets the rules for style that will improve your productivity - especially in teams, where understandability and consistency are critical. Coming from renowned UML expert Scott Ambler, the book furnishes a set of rules for modelling in the UML and describes a collection of standards and guidelines for creating effective UML diagrams that will be concise and easy to understand. It provides conventions for: Class diagrams; Timing Diagrams; Use case diagrams; Composite Structure Diagrams; Sequence diagrams; Interaction Overview Diagrams; Activity diagrams; Object diagrams; State machine diagrams; Package diagrams; Communication diagrams; Deployment diagrams and Component diagrams. The Elements of UMLTM 2.0 Style sets the rules for style that will improve your productivity.

This book constitutes the refereed joint proceedings of five international workshops held in conjunction with the 24th

International Conference on Conceptual Modeling, ER 2005, in Klagenfurt, Austria, in October 2005. The 40 revised full papers presented together with the abstracts of seven tutorials were carefully reviewed and selected from 102 submissions. The papers are organized in topical sections on best practices of UML, experience reports and new applications, model evaluation and requirements modeling, metamodeling and model driven development, positions in engineering agent oriented systems, agent oriented methodologies and conceptual modeling, agent communication and coordination, geographic information systems, spatial and spatio-temporal data representation, spatial relations, spatial queries, analysis and data mining, data modeling and visualisation, conceptual modeling approaches for e-business, information system models quality, and quality driven processes.

We would like to welcome you to the proceedings of the workshops held in c- junction with the 27th International Conference on Conceptual Modeling (ER 2008). While the ER main conference covers a wide spectrum of conceptual modeling research, increasingly complex real-world problems demand new p- spectives and activeresearchin new applications.The ER workshopsattempt to provideresearchers,students, andindustry professionalswitha forumto present and discuss emerging hot topics related to conceptual modeling. We received 13 excellent proposals for workshops to be held with ER 2008. We accepted the following seven based on peer reviews: 1. The Second International Workshop on Conceptual Modeling for Life S- ences Applications (CMLSA 2008), organized by Yi-Ping Phoebe Chen and Sven Hartmann. 2. The 5th International Workshopon Evolution and Change in Data Mana- ment (ECDM 2008), organized by Fabio Grandi. 3. The 4th International Workshop on Foundations and Practices of UML (FP-UML 2008), organized by Juan Trujillo and Andreas L. Opdahl. 4. The First International Workshop on Modeling Mobile Applications and Services (M2AS 2008), organized by Fernando Ferri, Patrizia Grifoni, and Maria Chiara Caschera. 5. The Second International Workshop on Requirements, Intentions and Goals in Conceptual Modeling (RIGiM 2008), organized by Colette Rolland, C- son Woo, and Camille Salinesi. 6. The Second International Workshop on Semantic and Conceptual Issues in Geographic Information Systems (SeCoGIS 2008), organized by Esteban Zim´ anyi and Christophe Claramunt. 7. The 5th International Workshop on Web Information Systems Modeling (WISM 2008), organized by Flavius Frasincar, Geert-Jan Houben, and Philippe Thiran.

The idea for this conference came from a meeting of the IFIP (International Federation for Information Processing) Technical Committee for Information Systems (TC8) in Guimares, Portugal in June 2005. Our goal is to build an IFIP forum among the different Information Systems Communities of TC8 dealing with the increasing important area of Enterprise Information Systems. In this particular meeting the committee members intensively discussed the innovative and unique characteristics of Enterprise Information Systems as scientific sub-discipline. Hence, in this meeting it was

decided by the TC8 members that the IFIP TC8 First International Conference on Research and Practical Issues of Enterprise Information Systems (CONFENIS 2006) would be held in April 2006 in Vienna, Austria. Dr. Li Xu (USA) and Dr. A Min Tjoa (IFIP TC8) were assigned to propose a concept for this conference in order to establish an IFIP platform for EIS researchers and practitioners in the field to share experience, and discussing opportunities and challenges. We are very pleased therefore to have this conference organised by the help of the Austrian Computer Society (OCG). OCG supports the idea of this conference due to the urgent need of research and dissemination of new techniques in this key area. We received 180 papers from more than 30 countries for CONFENIS and the Program Committee eventually selected xx papers or extended abstracts, making an acceptance rate of xx% of submitted papers. Each paper was thoroughly reviewed by at least two qualified reviewers.
"This book manages to convey the practical use of UML 2 in clear and understandable terms with many examples and guidelines. Even for people not working with the Unified Process, the book is still of great use. UML 2 and the Unified Process, Second Edition is a must-read for every UML 2 beginner and a helpful guide and reference for the experienced practitioner." --Roland Leibundgut, Technical Director, Zuehlke Engineering Ltd. "This book is a good starting point for organizations and individuals who are adopting UP and need to understand how to provide visualization of the different aspects needed to satisfy it. " --Eric Naiburg, Market Manager, Desktop Products, IBM Rational Software This thoroughly revised edition provides an indispensable and practical guide to the complex process of object-oriented analysis and design using UML 2. It describes how the process of OO analysis and design fits into the software development lifecycle as defined by the Unified Process (UP). UML 2 and the Unified Process contains a wealth of practical, powerful, and useful techniques that you can apply immediately. As you progress through the text, you will learn OO analysis and design techniques, UML syntax and semantics, and the relevant aspects of the UP. The book provides you with an accurate and succinct summary of both UML and UP from the point of view of the OO analyst and designer. This book provides Chapter roadmaps, detailed diagrams, and margin notes allowing you to focus on your needs Outline summaries for each chapter, making it ideal for revision, and a comprehensive index that can be used as a reference New to this edition: Completely revised and updated for UML 2 syntax Easy to understand explanations of the new UML 2 semantics More real-world examples A new section on the Object Constraint Language (OCL) Introductory material on the OMG's Model Driven Architecture (MDA) The accompanying website provides A complete example of a simple e-commerce system Open source tools for requirements engineering and use case modeling Industrial-strength UML course materials based on the book
This book gives a practical approach to modeling and analyzingcommunication protocols using UML 2. Network protocols

are alwayspresented with a point of view focusing on partial mechanisms andstarting models. This book aims at giving the basis needed foranybody to model and validate their own protocols. It follows apractical approach and gives many examples for the description andanalysis of well known basic network mechanisms forprotocols. The book firstly shows how to describe and validate the mainprotocol issues (such as synchronization problems, client-serverinteractions, layer organization and behavior, etc.) in an easy andunderstandable way. To do so, the book considers and presents themain traditional network examples (e.g. unidirectional flows,full-duplex com-munication, error recovering, alternating bit).Finally, it presents the outputs resulting from a few simulationsof these UML models. Other books usually only focus either on teaching UML or onanalyzing network protocols, however this book will allow readersto model network protocols using a new perspective and integratingthese two views, so facilitating their comprehension anddevelopment. Any university student studying in the field of computing science,or those working in telecommunications, embedded systems ornetworking will find this book a very useful addition.

Models are used in all kinds of engineering disciplines to abstract from the various details of the modelled entity in order to focus on a speci?c aspect. Like a blueprint in civil engineering, a software architecture providesan abstraction from the full software system's complexity. It allows software designers to get an overview on the system underdevelopmentandtoanalyzeitsproperties.Inthissense,modelsarethefoundation needed for software development to become a true engineering discipline. Especially when reasoning on a software system's extra-functional properties, its software architecture carries the necessary information for early, design-time analyses. These analyses take the software architecture as input and can be used to direct the design process by allowing a systematic evaluation of different design alternatives. For example, they can be used to cancel out decisions which would lead to architecture - signs whose implementation would not comply with extra-functionalrequirements like performance or reliability constraints. Besides such quality attributes directly visible to the end user, internal quality attributes, e.g., maintainability, also highly depend on the system's architecture. In addition to the above-mentioned technical aspects of software architecture m- els, non-technical aspects, especially project management-related activities, require an explicit software architecture model. The models are used as input for cost esti- tions, time-, deadline-, and resource planning for the development teams. They serve the project management activities of planning, executing, and controlling, which are necessary to deliver high-quality software systems in time and within the budget.

This book constitutes the thoroughly refereed proceedings of the 47th International Conference on Objects, Components, Models and Patterns, TOOLS EUROPE 2009, held in Zurich, Switzerland, in June/July 2009. TOOLS has played a major role in the spread of object-oriented and component technologies. It has now broadened its scope beyond the original

topics of object technology and component-based development to encompass all modern, practical approaches to software development. At the same time, TOOLS has kept its traditional spirit of technical excellence, its acclaimed focus on practicality, its well-proven combination of theory and applications, and its reliance on the best experts from academia and industry. The 17 regular papers and two short papers presented in this book, together with two invited papers, were carefully reviewed and selected from 67 submissions. The topics covered in this volume are reflection and aspects, models, theory, components, monitoring, and systems generation.

Concise and easy-to-understand guidelines and standards for creating UML 2.0 diagrams.

This book constitutes the refereed proceedings of the 7th International Conference on the Unified Modeling Language, UML 2004, held in Lisbon, Portugal, in October 2004. The 30 revised full papers presented together with summaries on the workshops and tutorials were carefully reviewed and selected from 135 technical paper submissions. The papers are organized in topical sections on metamodeling, aspects, profiles and extensions, OCL, model transformation, verification and model consistency, security, and methodology.

?Information Systems (IS) as a discipline draws on diverse areas including, technology, organisational theory, management and social science. The field is recognized as very broad and encompassing many themes and areas. However, the development of artefacts, or information systems development (ISD), in the broadest sense, is a central concern of the discipline. Significantly, ISD impacts on the organisational and societal contexts through the use of the artefacts constructed by the development. Today, that impact also needs to be evaluated in terms of its effects on the environment. Sustainable, or "green," IT is a catch-all term used to describe the development, manufacture, management, use and disposal of ICT in a way that minimizes damage to the environment. As a result, the term has many different meanings, depending on the role assumed in the life span of the ICT artefact. The theme of the proposed work is to critically examine the whole range of issues around ISD from the perspective of sustainability. Sustainable IT is an emerging theme in academic research and industry practice in response to an individual concern for the environment and the embryonic regulatory environments being enacted globally to address the environmental impact of ICT. In this work we intend to bring together in one volume the diverse research around the development of sustainable IS.

This thoroughly revised second edition is based upon the authors' successful training courses, and gives the reader a quick, focused tour through a proven object-oriented analysis and design process. It introduces and explains the need-to-know concepts and key elements of both Unified Modeling Language (UML) and the Unified Process (UP). The book allows the reader to get up to speed on successful techniques that can be immediately applied. The structure of the book aims to give the reader as clear and uncomplicated a guide as possible. It takes the reader through an introduction to

UML, an int.

This textbook mainly addresses beginners and readers with a basic knowledge of object-oriented programming languages like Java or C#, but with little or no modeling or software engineering experience – thus reflecting the majority of students in introductory courses at universities. Using UML, it introduces basic modeling concepts in a highly precise manner, while refraining from the interpretation of rare special cases. After a brief explanation of why modeling is an indispensable part of software development, the authors introduce the individual diagram types of UML (the class and object diagram, the sequence diagram, the state machine diagram, the activity diagram, and the use case diagram), as well as their interrelationships, in a step-by-step manner. The topics covered include not only the syntax and the semantics of the individual language elements, but also pragmatic aspects, i.e., how to use them wisely at various stages in the software development process. To this end, the work is complemented with examples that were carefully selected for their educational and illustrative value. Overall, the book provides a solid foundation and deeper understanding of the most important object-oriented modeling concepts and their application in software development. An additional website offers a complete set of slides to aid in teaching the contents of the book, exercises and further e-learning material. Presents modeling approaches that can be performed in SysML and other modeling languages This book combines the emerging discipline of systems architecting with model-based approaches using SysML. The early chapters of the book provide the fundamentals of systems architecting; discussing what systems architecting entails and how it benefits systems engineering. Model-based systems engineering is then defined, and its capabilities to develop complex systems on time and in a feasible quality are discussed. The remainder of the book covers important topics such as: architecture descriptions; architecture patterns; perspectives, viewpoints, views and their relation to system architecture; the roles of a system architect, their team, and stakeholders; systems architecting processes; agile approaches to systems architecting; variant modeling techniques; architecture frameworks; and architecture assessment. The book's organization allows experts to read the chapters out of sequence. Novices can read the chapters sequentially to gain a systematic introduction to system architecting. Model-Based System Architecture: Provides comprehensive coverage of the Functional Architecture for Systems (FAS) method created by the authors and based on common MBSE practices Covers architecture frameworks, including the System of Systems, Zachman Frameworks, TOGAF®, and more Includes a consistent example system, the "Virtual Museum Tour" system, that allows the authors to demonstrate the systems architecting concepts covered in the book Model-Based System Architecture is a comprehensive reference for system architects and systems engineers in technology companies. This book will also serve as a reference to students and researchers interested in functional architectures. Tim Weilkiens is the CEO at the German consultancy oose Innovative

Informatik and co-author of the SysML specification. He has introduced model-based systems engineering to a variety of industry sectors. He is author of several books about modeling and the MBSE methodology SYSMOD. Jesko G. Lamm is a Senior Systems Engineer at Bernafon, a Swiss manufacturer for hearing instruments. With Tim Weilkiens, Jesko G. Lamm founded the Functional Architectures working group of the German chapter of INCOSE. Stephan Roth is a coach, consultant, and trainer for systems and software engineering at the German consultancy oose Innovative Informatik. He is a state-certified technical assistant for computer science from Physikalisch-Technische Lehranstalt (PTL) Wedel and a certified systems engineer (GfSE)®- Level C. Markus Walker works at Schindler Elevator in the research and development division as elevator system architect. He is an INCOSE Certified Systems Engineering Professional (CSEP) and is engaged in the committee of the Swiss chapter of INCOSE.

A guide to using UML describes major UML diagrams, their creation, and how to decipher them.