

Object Oriented Software Engineering

Object-Oriented Software
Object-oriented Software and Engineering
Object-oriented Software Construction
Object-oriented Software Engineering with C++
Object-oriented Software Engineering
Designing Object-oriented Software
Object-oriented Software Engineering with Eiffel
Scientific Software Design
Object-oriented Software Engineering
Classical and Object-oriented Software Engineering with UML and C++
Object-oriented Software Engineering
Project-based Software Engineering
Exam Prep for: Object-Oriented Software Engineering
Using Object-oriented and Classical Software Engineering
OBJECT-ORIENTED SOFTWARE ENGINEERING
Classical and Object-oriented Software Engineering
Object-Oriented Software Engineering
Object-oriented Software Engineering
Reflection and Software Engineering
Object Oriented Software Engineering
Conquering Complex And Changing Systems
Object-Oriented and Classical Software Engineering
Quantitative Approaches in Object-oriented Software Engineering
Essays on Object-oriented Software Engineering
Object-Oriented Metrics in Practice
Object-Oriented Software Engineering
Object Oriented Software Engineering
Object-oriented Software Construction
Object-oriented Construction Handbook
Software Engineering: A Hands-On Approach
Object-Oriented Software Engineering: An Agile Unified Methodology
Object-Oriented Software

Bookmark File PDF Object Oriented Software Engineering

Engineering: Practical Software Development using UML and JavaSoftware Engineering (Sie) 7EObject-Oriented Software Engineering Using UML, Patterns, and Java: Pearson New International EditionTesting Object-Oriented SoftwareApplications and Approaches to Object-Oriented Software Design: Emerging Research and OpportunitiesObject-Oriented Software Engineering: An Agile Unified MethodologyObject-oriented Software EngineeringObject-oriented Software Development Using JavaSoftware EngineeringGrowing Object-Oriented Software, Guided by Tests

Object-Oriented Software

An indispensable resource for anyone working with Eiffel, this up-to-date guide provides full coverage of the most recent version of the language, focusing on Eiffel's practical use in the development of large, mission-critical software systems. In addition to a comprehensive description of Eiffel's syntax and semantics, you will find in-depth information on style guides, analysis and design, design patterns, and validation and testing. Descriptions and comparisons of available compilers and libraries will help you decide which Eiffel tools best fit your development needs. The book even includes an Eiffel resource guide. The book's most notable feature is its three large-scale case studies that demonstrate Eiffel in action, illustrating implementation techniques and showcasing Eiffel's power and effectiveness in three different realms: the MIS world, the embedded systems/telecommunications world, and the numeric

Bookmark File PDF Object Oriented Software Engineering

world. By reading this book, you will not only obtain a knowledge of the mechanics of Eiffel programming, but you will also come away with an understanding of Eiffel's role in the field of object-oriented technology and a sense of the language's strong potential in large software development. 0201633817B04062001

Object-oriented Software and Engineering

Object-Oriented Software Engineering is written for both the traditional one-semester and the newer two-semester software engineering curriculum. Part I covers the underlying software engineering theory, while Part II presents the more practical life cycle, workflow by workflow. The text is intended for the substantial object-oriented segment of the software engineering market. It focuses exclusively on object-oriented approaches to the development of large software systems that are the most widely used. Text includes 2 running case studies, expanded coverage of agile processes and open-sour.

Object-oriented Software Construction

Object-Oriented Software Engineering: An Agile Unified Methodology by David Kung presents a step-by-step methodology that integrates modeling and design, UML, patterns, test-driven development, quality assurance, configuration management, and agile principles throughout the life cycle. The overall approach is casual and easy to follow, with many practical examples that show the theory at work. The

Bookmark File PDF Object Oriented Software Engineering

author uses his experiences as well as real-world stories to help the reader understand software design principles, patterns, and other software engineering concepts. The book also provides stimulating exercises that go far beyond the type of question that can be answered by simply copying portions of the text.

Object-oriented Software Engineering with C++

This is a textbook for a course in object-oriented software engineering at advanced undergraduate and graduate levels, as well as for software engineers. It contains more than 120 exercises of diverse complexity. The book discusses fundamental concepts and terminology on object-oriented software development, assuming little background on software engineering, and emphasizes design and maintenance rather than programming. It also presents up-to-date and easily understood methodologies and puts forward a software life cycle model which explicitly encourages reusability during software development and maintenance.

Object-oriented Software Engineering

Designing Object-oriented Software

Venturing beyond C++ programming, this text shows how to engineer software products using object-oriented principles. It covers gathering requirements,

Bookmark File PDF Object Oriented Software Engineering

specifying objects, object verification, defining relations between objects, translating object design into code, object testing, and software maintenance.

Object-oriented Software Engineering with Eiffel

Presents a novel metrics-based approach for detecting design problems in object-oriented software. Introduces an important suite of detection strategies for the identification of different well-known design flaws as well as some rarely mentioned ones.

Scientific Software Design

This volume aims to study how practicing software developers, in industrial as well as academic environments, can use object technology to improve the quality of the software they produce. It includes topics on concurrency and Internet programming.

Object-oriented Software Engineering

Classical and Object-oriented Software Engineering with UML and C++

Software -- Software Engineering.

Object-oriented Software Engineering

This comprehensive and well-written book presents

Bookmark File PDF Object Oriented Software Engineering

the fundamentals of object-oriented software engineering and discusses the recent technological developments in the field. It focuses on object-oriented software engineering in the context of an overall effort to present object-oriented concepts, techniques and models that can be applied in software estimation, analysis, design, testing and quality improvement. It applies unified modelling language notations to a series of examples with a real-life case study. The example-oriented approach followed in this book will help the readers in understanding and applying the concepts of object-oriented software engineering quickly and easily in various application domains. This book is designed for the undergraduate and postgraduate students of computer science and engineering, computer applications, and information technology. KEY FEATURES : Provides the foundation and important concepts of object-oriented paradigm. Presents traditional and object-oriented software development life cycle models with a special focus on Rational Unified Process model. Addresses important issues of improving software quality and measuring various object-oriented constructs using object-oriented metrics. Presents numerous diagrams to illustrate object-oriented software engineering models and concepts. Includes a large number of solved examples, chapter-end review questions and multiple choice questions along with their answers.

Project-based Software Engineering

This book is based on object-oriented techniques

Bookmark File PDF Object Oriented Software Engineering

applied to software engineering. Employing the latest technologies such as UML, Patterns, and Java, Bernd Bruegge and Allen H. Dutoit offer a cohesive, class-tested presentation of object-oriented software engineering in a step-by-step format based on ten years of teaching and real-world software engineering experience. This text teaches practical experience in developing complex software appropriate for software engineering project courses, as well as industry R & D practitioners. The reader benefits from timely exposure to state-of-the-art tools and methods.

Exam Prep for: Object-Oriented Software Engineering Using

The authors analyze how the structure of a package determines its developmental complexity according to such measures as bug search times and documentation information content. The work presents arguments for why these issues impact solution cost and time more than does scalable performance. The final chapter explores the question of scalable execution and shows how scalable design relates to scalable execution. The book's focus is on program organization, which has received considerable attention in the broader software engineering community, where graphical description standards for modeling software structure and behavior have been developed by computer scientists. These discussions might be enriched by engineers who write scientific codes. This book aims to bring such scientific programmers into discussion with computer scientists. The authors do so by

Bookmark File PDF Object Oriented Software Engineering

introducing object-oriented software design patterns in the context of scientific simulation.

Object-oriented and Classical Software Engineering

Object-Oriented Software Engineering: An Agile Unified Methodology, presents a step-by-step methodology - that integrates Modeling and Design, UML, Patterns, Test-Driven Development, Quality Assurance, Configuration Management, and Agile Principles throughout the life cycle. The overall approach is casual and easy to follow, with many practical examples that show the theory at work. The author uses his experiences as well as real-world stories to help the reader understand software design principles, patterns, and other software engineering concepts. The book also provides stimulating exercises that go far beyond the type of question that can be answered by simply copying portions of the text.

OBJECT-ORIENTED SOFTWARE ENGINEERING

Object-oriented programming (OOP) has been the leading paradigm for developing software applications for at least 20 years. Many different methodologies, approaches, and techniques have been created for OOP, such as UML, Unified Process, design patterns, and eXtreme Programming. Yet, the actual process of building good software, particularly large, interactive, and long-lived software, is still emerging. Software

Bookmark File PDF Object Oriented Software Engineering

engineers familiar with the current crop of methodologies are left wondering, how does all of this fit together for designing and building software in real projects? This handbook from one of the world's leading software architects and his team of software engineers presents guidelines on how to develop high-quality software in an application-oriented way. It answers questions such as: * How do we analyze an application domain utilizing the knowledge and experience of the users? * What is the proper software architecture for large, distributed interactive systems that can utilize UML and design patterns? * Where and how should we utilize the techniques and methods of the Unified Process and eXtreme Programming? This book brings together the best of research, development, and day-to-day project work. "The strength of the book is that it focuses on the transition from design to implementation in addition to its overall vision about software development."
-Bent Bruun Kristensen, University of Southern Denmark, Odense

Classical and Object-oriented Software Engineering

Object-Oriented Software Engineering

Object-oriented Software Engineering

Jia (software engineering, DePaul University) helps readers develop skills in designing software, and

Bookmark File PDF Object Oriented Software Engineering

especially in writing object-oriented programs using Java. The text provides broad coverage of object-oriented technology, including object-oriented modeling using the Unified Modeling Language (UML), object-oriented design using design patterns, and object-oriented programming using Java. This second edition offers expanded coverage of design patterns, enhanced material on UML, and a new introduction to the iterative software development process made popular by extreme programming. Learning features include chapter summaries, exercises, and projects.

Reflection and Software Engineering

For courses in Software Engineering, Software Development, or Object-Oriented Design and Analysis at the Junior/Senior or Graduate level. This text can also be utilized in short technical courses or in short, intensive management courses. Shows students how to use both the principles of software engineering and the practices of various object-oriented tools, processes, and products. Using a step-by-step case study to illustrate the concepts and topics in each chapter, Bruegge and Dutoit emphasize learning object-oriented software engineering through practical experience: students can apply the techniques learned in class by implementing a real-world software project. The third edition addresses new trends, in particular agile project management (Chapter 14 Project Management) and agile methodologies (Chapter 16 Methodologies).

Object Oriented Software Engineering

Conquering Complex And Changing Systems

Test-Driven Development (TDD) is now an established technique for delivering better software faster. TDD is based on a simple idea: Write tests for your code before you write the code itself. However, this "simple" idea takes skill and judgment to do well. Now there's a practical guide to TDD that takes you beyond the basic concepts. Drawing on a decade of experience building real-world systems, two TDD pioneers show how to let tests guide your development and "grow" software that is coherent, reliable, and maintainable. Steve Freeman and Nat Pryce describe the processes they use, the design principles they strive to achieve, and some of the tools that help them get the job done. Through an extended worked example, you'll learn how TDD works at multiple levels, using tests to drive the features and the object-oriented structure of the code, and using Mock Objects to discover and then describe relationships between objects. Along the way, the book systematically addresses challenges that development teams encounter with TDD—from integrating TDD into your processes to testing your most difficult features. Coverage includes

- Implementing TDD effectively: getting started, and maintaining your momentum throughout the project
- Creating cleaner, more expressive, more sustainable code
- Using tests to stay relentlessly focused on sustaining quality
- Understanding how TDD, Mock Objects, and Object-Oriented Design come together in the context of a real software development project

Bookmark File PDF Object Oriented Software Engineering

Using Mock Objects to guide object-oriented designs
Succeeding where TDD is difficult: managing complex test data, and testing persistence and concurrency

Object-Oriented and Classical Software Engineering

This book has been written to communicate the complexity of software engineering, a field that is on the rise. Braude has combined practical industrial experience with up-to-date academic experience to give the reader a feel for the complexity and important issues of real-world development. A longitudinal case study using IEEE standards is implemented throughout the book, along with many other examples, which enables the reader to understand the implications of quality factors, proper requirements documents, appropriate design, and appropriate project management techniques.

Quantitative Approaches in Object-oriented Software Engineering

Examines object-oriented methods, practices, terminology, and concepts

Essays on Object-oriented Software Engineering

This book presents the state of the art of research and development of computational reflection in the context of software engineering. Reflection has attracted considerable attention recently in software

Bookmark File PDF Object Oriented Software Engineering

engineering, particularly from object-oriented researchers and professionals. The properties of transparency, separation of concerns, and extensibility supported by reflection have largely been accepted as useful in software development and design; reflective features have been included in successful software development technologies such as the Java language. The book offers revised versions of papers presented first at a workshop held during OOPSLA'99 together with especially solicited contributions. The papers are organized in topical sections on reflective and software engineering foundations, reflective software adaptability and evolution, reflective middleware, engineering Java-based reflective languages, and dynamic reconfiguration through reflection.

Object-Oriented Metrics in Practice

This text provides an introduction to the process of software engineering. The revision concentrates on updating the book to reflect the most current trends and innovations in the field. The Universal Modeling Language (UML) has become an industry standard and now permeates this new edition. In this text, it is used for object-oriented analysis and design as well as when diagrams depict objects and their interrelationships. Design patterns, frameworks and software architecture have also become a popular topic in the field of software engineering and are part of a new chapter on reuse, portability, and inoperability. The inoperability material includes sections on such hot topics as OLE, COM, and CORBA.

Bookmark File PDF Object Oriented Software Engineering

Some material from the 3rd edition has been reorganized into a new chapter on planning and estimating, including feature points and COCOMO II. While the text has been updated, the traditional features which have defined the previous three editions of Schach's book have been retained. These include a balanced coverage of the object-oriented model along with the classical model (as reflected in the title) and an emphasis on metrics. The special considerations of object-oriented life-cycle models, object-oriented analysis, and object-oriented design are also retained in this edition.

Object-Oriented Software Engineering

Object Oriented Software Engineering

Object-oriented Software Construction

Addressing various aspects of object-oriented software techniques with respect to their impact on testing, this text argues that the testing of object-oriented software is not restricted to a single phase of software development. The book concentrates heavily on the testing of classes and of components or sub-systems, and a major part is devoted to this subject. C++ is used throughout this book that is intended for software practitioners, managers, researchers, students, or anyone interested in object-oriented technology and its impacts throughout the software engineering life-cycle.

Object-oriented Construction Handbook

A comprehensive description is provided of four areas of quantitative approaches in object-oriented software engineering: metrics collection, quality assessment, metrics validation, and process management.

Software Engineering: A Hands-On Approach

Object-Oriented Software Engineering is written for both the traditional one-semester and the newer two-semester software engineering curriculum. Part I covers the underlying software engineering theory, while Part II presents the more practical life cycle, workflow by workflow. The text is intended for the substantial object-oriented segment of the software engineering market. It focuses exclusively on object-oriented approaches to the development of large software systems that are the most widely used. Text includes 2 running case studies, expanded coverage of agile processes and open-source development.

Object-Oriented Software Engineering: An Agile Unified Methodology

Project-Based Software Engineering is the first book to provide hands-on process and practice in software engineering essentials for the beginner. The book presents steps through the software development life cycle and two running case studies that develop as the steps are presented. Running parallel to the process presentation and case studies, the book

Bookmark File PDF Object Oriented Software Engineering

supports a semester-long software development project. This book focuses on object-oriented software development, and supports the conceptualization, analysis, design and implementation of an object-oriented project. It is mostly language-independent, with necessary code examples in Java. A subset of UML is used, with the notation explained as needed to support the readers' work. Two running case studies a video game and a library check out system show the development of a software project. Both have sample deliverables and thus provide the reader with examples of the type of work readers are to create. This book is appropriate for readers looking to gain experience in project analysis, design implementation, and testing.

Object-Oriented Software Engineering: Practical Software Development using UML and Java

This book covers the essential knowledge and skills needed by a student who is specializing in software engineering. Readers will learn principles of object orientation, software development, software modeling, software design, requirements analysis, and testing. The use of the Unified Modelling Language to develop software is taught in depth. Many concepts are illustrated using complete examples, with code written in Java.

Software Engineering (Sie) 7E

& Quot;Object Orientation and Classical Software

Bookmark File PDF Object Oriented Software Engineering

Engineering, Sixth Edition, by Stephen R. Schach has been revised to incorporate the Unified Process throughout, making the text more practical and modern. UML coverage has also been enhanced, and the references have been updated to keep students abreast of what is going on in software engineering today. The sixth edition retains its student-friendly approach and remains a standard in the field for learning software engineering. & quot;--BOOK JACKET.

Object-Oriented Software Engineering Using UML, Patterns, and Java: Pearson New International Edition

Testing Object-Oriented Software

This book covers the essential knowledge and skills needed by a student who is specializing in software engineering. Readers will learn principles of object orientation, software development, software modeling, software design, requirements analysis, and testing. The use of the Unified Modelling Language to develop software is taught in depth. Many concepts are illustrated using complete examples, with code written in Java.

Applications and Approaches to Object-Oriented Software Design: Emerging Research and Opportunities

Software -- Software Engineering.

Object-Oriented Software Engineering: An Agile Unified Methodology

In today's modernized environment, a growing number of software companies are changing their traditional engineering approaches in response to the rapid development of computing technologies. As these businesses adopt modern software engineering practices, they face various challenges including the integration of current methodologies and contemporary design models and the refactoring of existing systems using advanced approaches. Applications and Approaches to Object-Oriented Software Design: Emerging Research and Opportunities is a pivotal reference source that provides vital research on the development of modern software practices that impact maintenance, design, and developer productivity. While highlighting topics such as augmented reality, distributed computing, and big data processing, this publication explores the current infrastructure of software systems as well as future advancements. This book is ideally designed for software engineers, IT specialists, data scientists, business professionals, developers, researchers, students, and academicians seeking current research on contemporary software engineering methods.

Object-oriented Software Engineering

This textbook provides a progressive approach to the teaching of software engineering. First, readers are introduced to the core concepts of the object-oriented methodology, which is used throughout the book to

Bookmark File PDF Object Oriented Software Engineering

act as the foundation for software engineering and programming practices, and partly for the software engineering process itself. Then, the processes involved in software engineering are explained in more detail, especially methods and their applications in design, implementation, testing, and measurement, as they relate to software engineering projects. At last, readers are given the chance to practice these concepts by applying commonly used skills and tasks to a hands-on project. The impact of such a format is the potential for quicker and deeper understanding. Readers will master concepts and skills at the most basic levels before continuing to expand on and apply these lessons in later chapters.

Object-oriented Software Development Using Java

Software Engineering

Growing Object-Oriented Software, Guided by Tests

For courses in Software Engineering, Software Development, or Object-Oriented Design and Analysis at the Senior or Graduate level. This text explores the theoretical foundations of software engineering and the principles and practices of various object-oriented tools, processes, and products.

Bookmark File PDF Object Oriented Software Engineering

[ROMANCE](#) [ACTION & ADVENTURE](#) [MYSTERY & THRILLER](#) [BIOGRAPHIES & HISTORY](#) [CHILDREN'S](#) [YOUNG ADULT](#) [FANTASY](#) [HISTORICAL FICTION](#) [HORROR](#) [LITERARY FICTION](#) [NON-FICTION](#) [SCIENCE FICTION](#)