Essays On Object Oriented Software Engineering

Third International Conference on Software Reuse: Advances in Software ReusabilityEssays on Object-oriented Software EngineeringObject-oriented Software EngineeringObject-oriented MethodsScrumban - Essays on Kanban Systems for Lean Software DevelopmentJournal of Object-oriented ProgrammingSoftware EngineeringObject-oriented Systems Analysis and DesignObject-oriented Technology For Database And Software SystemsOBJECT-ORIENTED SOFTWARE ENGINEERINGTesting Object-Oriented SoftwareObject-oriented Analysis and Design with ApplicationsObject MagazineObject-oriented System DevelopmentObject ThinkingObject-Oriented Design MeasurementUnderstanding Object-oriented Software EngineeringClassical and Object-oriented Software Engineering with UML and JavaObject Oriented Analysis, Design and TestingClassical and Object-oriented Software EngineeringProgramming on Purpose IIITechnology of Object-oriented Languages and Systems, TOOLS 11The Well-tempered ObjectComputer LanguageBest of BoochFrom Object-Orientation to Formal MethodsObject-oriented Development Process and MetricsObject Technology TransferTesting Object-Oriented SoftwareConfiguration Management Based on Software Component Locality and System StructureCognitive PatternsMusings of a Modern Man: Essays and Research PapersObject-oriented Programming with C++ and SmalltalkOnline Education Using Learning ObjectsObject-oriented MetricsA Book of Object-oriented

KnowledgeAsia-Pacific Software Engineering ConferenceBooktwo of Object-oriented KnowledgeSoftware QualityObject Expo Europe Conference Proceedings

Third International Conference on Software Reuse: Advances in Software Reusability

"Software Engineering" describes the current state-of-the-art practice of software engineering, beginning with an overview of current issues and focusing on the engineering of large complex systems. The text illustrates the phases of the software development life cycle: requirements, design, implementation, testing and maintenance.

Essays on Object-oriented Software Engineering

"Based on a thorough study of measurement theory, category theory, and a strong mathematic foundation, Scott Whitmire presents his own formal model of objects - a ground-breaking development for object-oriented design measurement. Using this model, or any other formal model, you can create a custom metric for any design characteristic you can describe through careful observation. This indispensable book guides you through the development process for nine unique design characteristics, including a controversial new "quality of abstraction"

characteristic, which combines Sufficiency, Completeness, and Cohesion to help you determine the extent to which components include all the features required of your design."--Page 4 of cover.

Object-oriented Software Engineering

This collection of material is assembled from the latest research in the field of object technology education and training. The contributors are mainly from industry, and include experts from the USA, the UK and Europe.

Object-oriented Methods

Scrumban - Essays on Kanban Systems for Lean Software Development

This work examines software quality assurance in practice and includes standards and models.

Journal of Object-oriented Programming

In OBJECT THINKING, esteemed object technologist David West contends that the mindset makes the programmer—not the tools and techniques. Delving into the history, philosophy, and even politics of object-oriented programming, West reveals how the best programmers rely on analysis and conceptualization—on thinking—rather than formal process and methods. Both provocative and pragmatic, this book gives form to what's primarily been an oral tradition among the field's revolutionary thinkers—and it illustrates specific object-behavior practices that you can adopt for true object design and superior results. Gain an in-depth understanding of: Prerequisites and principles of object thinking. Object knowledge implicit in eXtreme Programming (XP) and Agile software development. Object conceptualization and modeling. Metaphors, vocabulary, and design for object development. Learn viable techniques for: Decomposing complex domains in terms of objects. Identifying object relationships, interactions, and constraints. Relating object behavior to internal structure and implementation design. Incorporating object thinking into XP and Agile practice.

Software Engineering

Software -- Software Engineering.

Object-oriented Systems Analysis and Design

This collection of essays drawn from Plauger's popular "Programming on Purpose" column in the magazine Computer Language, focuses on the technology of writing computer software. Plauger's style is clear without being simplistic, reducing complex themes to bite-size chunks. KEY TOPICS: Covers a number of important technical themes such as computer arithmetic, approximating math functions, human perception and artificial intelligence, encrypting data and clarifying documentation.

Object-oriented Technology For Database And Software Systems

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

OBJECT-ORIENTED SOFTWARE ENGINEERING

Object orientation has become a "must know" subject for managers, researchers, and software practitioners interested in the design, evolution, reuse and management of efficient software components.The book contains technical papers reflecting both theoretical and practical contributions from researchers in the field of object-oriented (OO) databases and software engineering systems. The book identifies actual and potential areas of integration of OO and database

technologies, current and future research directions in software methodologies, and reflections about the OO paradigm.In providing current research and relevant information about this promising and rapidly growing field of object-oriented databases and software engineering systems, this book is invaluable to research scientists, practitioners, and graduate students working in the areas of databases and software engineering.

Testing Object-Oriented Software

These papers on object-oriented analysis and design cover: overviews of the object-oriented paradigm; methodologies; requirements analysis for applications; OO design and design patterns; and testing and maintenance of OO applications.

Object-oriented Analysis and Design with Applications

Object Magazine

'E-learning is integral to on-site education institutions worldwide, and the rapid explosion of interest in the subject means that this timely, cutting-edge book will be an instant and indispensable resource. Among educators, the development of

reusable learning objects made accessible via the internet is ever more important to teaching and learning. This book provides a comprehensive look at a state-of-the-art online education, and presents advice on the creation, adaptation and implementation of learning objects and metadata. Including articles written by some of the leading innovators in the field, this book takes the reader through: designing effective learning objects; creating learning objects; transforming existing content into reusable learning objects; building a metadata management system. This book will be essential reference material for learning technologists, course developers at learning institutions, postgraduate students, teachers and learners in the field of e-learning.'

Object-oriented System Development

Object Thinking

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 Design Measurement

With this book, software engineers, project managers, and tool builders will be able to better understand the role of analysis and design in the object-oriented (OO) software development process. This book presents a minimum set of notions and shows the reader how to use these notions for OO software construction. The emphasis is on development principles and implementation.

Understanding Object-oriented Software Engineering

Designed for software professionals who are concerned about the success of their object-oriented projects, this volume covers all aspects of the Booch method and how a complete method must address a model's notation and semantics as well as a proccess for creating that model

Classical and Object-oriented Software Engineering with UML and Java

In this text, Dennis de Champeaux clearly presents the practical basics of each phase of the software development life-cycle using an object-oriented analysis and design process. A detailed home heating system is effectively used as an example throughout the text to instruct project managers, software developers, and researchers how to transform from an analysis via design into an implementation. The author uses state-of-the-art techniques to identify detailed software development micro processes to help manage projects and schedules. In addition, discussions are included on following micro processes, how effort prediction can be done, and how effort metrics can be applied to systematically collect historical data.

Object Oriented Analysis, Design and Testing

The three keynote addresses cover the new for new software technology, principles of software evolution, and auditing informal software testing and analysis processes. Others of the 41 papers consider a design metric for module coupling, a class testing technique based on data binding, safety analysis using colored Petri nets, a dynamic extension for specifying distributed systems, designing and implementing dynamically reconstructing system software, a dialog-oriented user interface generation mechanism, real-time system design tools based on a real-time object model, and other topics. No subject index. Annotation

copyrighted by Book News, Inc., Portland, OR.

Classical and Object-oriented Software Engineering

Programming on Purpose III

Technology of Object-oriented Languages and Systems, TOOLS 11

Since the previous edition of this popular and comprehensive book was published, there have been massive changes in the field of object technology. This book has been fully revised and updated to reflect the newest technologies and methodologies, including extensive coverage of middleware, components, Java & UML. If you are a developer or manager needing to succeed with objects, this book will give you a full understanding of the key concepts, benefits and pitfalls - plus what technologies and tools are available and how to evaluate them. It offers invaluable insights into the philosophy and real-world practice of today' s leading object-oriented techniques and products. Major features of this edition: detailed chapter covering middleware and migration strategies chapter describing best

practice for analysis and design, with in-depth focus on architecture and patterns plus a concise presentation of the Catalysis method for component based development revised coverage of requirements, featuring detailed description of the SOMA approach coverage of Java, in addition to other object-oriented programming languages Plus:- significantly revised coverage of object-oriented databases to address new and increasingly mature products- review of processes and project management including RUP and OPEN Process, and guidance on testing and UI design- new appendices summarizing the UML notation and background survey of 50 object oriented methods- self-test questions and model answers on accompanying web-site: www. trireme.com

The Well-tempered Object

This book describes the design goals and language features of object-oriented languages without viewing them from the perspective of any particular language. Covers key object-oriented principles — date abstraction, inheritance, polymorphism, and dynamic binding in a language independent discussion that focuses on the purpose of each feature.

Computer Language

Designed for technical managers and software engineers whose focus is on methods, procedures, and management -- rather than on coding -- this comprehensive discussion of software development using an Object-Oriented approach describes, in detail, a second generation, full life-cycle Object- Oriented methodology (MOSES) -- taking the developer from the initial user requirements and business planning through to implementation, testing, maintenance, and future enhancements. It covers the object-oriented paradigm; object-oriented development; MOSES notation; MOSES product and process lifecycles; MOSES lifecycle activities; a case study using MOSES; project management and commercial adoption of OOSE; and object-oriented "metrics."

Best of Booch

From Object-Orientation to Formal Methods

This edited collection of articles from Computer Music Journal provides a timely and convenient source of tutorials on OOP languages and software design techniques and surveys a wide range of existing applications of this technology to music and digital signal processing. Object-oriented programming (OOP) is perhaps the most important new software engineering technology of the past decade and promises

to be a key factor in much of the software of the 1990s. This edited collection of articles from Computer Music Journal provides a timely and convenient source of tutorials on OOP languages and software design techniques and surveys a wide range of existing applications of this technology to music and digital signal processing. Included are the popular OOP languages LISP, Smalltalk-80, and Objective-C, and applications such as music description and composition, real-time performance, and digital signal processing.Contents. Introduction. Tutorials and Technology. Machine Tongues VIII: The Design of a Smalltalk Music System, Glenn E. Krasner. Machine Tongues IX: Object-Oriented Programming, Henry Lieberman. Machine Tongues XI: Object-Oriented Software Design, Stephen Pope. Music Representation and Processing Tools. Flavors Band: A Language for Specifying Musical Style, Christopher Fry. FORMES: Composition and Scheduling of Processes, Xavier Roder and Pierre Cointe. An Introduction to the MODE System - A Musical Object Development Environment, Stephen Pope. An Overview of the Sound and Music Kits for the NeXT Computer, David Jaffe and Lee Boynton. Composition Systems. The Kyma/Platypus Computer Music Workstation, Carla Scaletti. An Introduction to the Creation Station, Henry Flurry. TTrees: A Tool for the Compositional Environment, Glendon Diener. Signal Processing Systems. Javelina: An Environment for Digital Signal Processor Software Development, Kurt Hebel. Virtual Digital Signal Processing in an ObjectOriented System, David Mellinger, Guy Garnett, and Bernard Mont-Reynaud.

Object-oriented Development Process and Metrics

This book is a collection of essays and research papers on a wide variety of subjects, including philosophy, computers, drama, poetry, literature, history, art, music, theology, networking, mathematics, business economics, transmission media and databases.

Object Technology Transfer

This comprehensive and well-written book presents 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.

Testing Object-Oriented Software

Object-oriented programming increases software reusability, extensibility, interoperability, and reliability. Software testing is necessary to realize these benefits. Software testing aims to uncover as many programming errors as possible at a minimum cost. A major challenge to the software engineering community remains how to reduce the cost and improve the quality of software testing. The requirements for testing object-oriented programs differ from those for testing conventional programs. Testing Object-Oriented Software illustrates these differences and discusses object-oriented software testing problems, focusing on the difficulties and challenges testers face. The book provides a general framework for class- and system-level testing and examines object-oriented design criteria and high testability metrics. It offers object-oriented testing techniques, ideas and methods for unit testing, and object-oriented program integration-testing strategy.

Readers are shown how they can drastically reduce regression test costs, presented with steps for object-oriented testing, and introduced to object-oriented test tools and systems. In addition to software testing problems, the text covers various test methods developers can use during the design phase to generate programs with good testability. The book's intended audience includes object-oriented program testers, program developers, software project managers, and researchers working with object-oriented testing.

Configuration Management Based on Software Component Locality and System Structure

Aiming to provide a comprehensive introduction to object-orientation, this book places an emphasis on analysis and design and presents a coherent methodology. It includes a chapter on software engineering and uses a running example to illustrate the concepts of object-orientation.

Cognitive Patterns

Musings of a Modern Man: Essays and Research Papers

After Ole-Johan's retirement at the beginning of the new millennium, some of us had thought and talked about making a "Festschrift" in his honor. When Donald Knuth took the initiative by sending us the ?rst contribution, the p- cess began to roll! In early 2002 an editing group was formed, including Kristen Nygaard, who had known Ole-Johan since their student days, and with whom he had developed the Simula language. Then we invited a number of prominent researchers familiar with Ole-Johan to submit contributions for a book hon- ing Ole-Johan on the occasion of his 70th birthday. Invitees included several members of the IFIP 2. 3 working group, a forum that Ole-Johan treasured and enjoyedparticipating in throughouthis career. In spite of the shortdeadline, the response to the invitations was overwhelmingly positive. The original idea was to complete the book rather quickly to make it a gift he could read and enjoy, because by then he had had cancer for three years, and his health was gradually deteriorating. Kristen had been regularly visiting Ole-Johan,who was in the hospitalat that time, and they were workingon their Turing award speech. Ole-Johan was grati?ed to hear about the contributions to this book, but modestly expressed the feeling that there was no special need to undertake a book project on his behalf. Peacefully accepting his destiny, Ole- Johan died on June 29, 2002.

Object-oriented Programming with C++ and Smalltalk

Online Education Using Learning Objects

Object-oriented Metrics

A Book of Object-oriented Knowledge

Presents the concepts and terminology of cognitive patterns and modeling and explains the uniqueness of cognitive patterns as an approach in modeling business systems and processes.

Asia-Pacific Software Engineering Conference

Evolutionary in approach, this book explores informatino systems development--both analysis and design--using an object-oriented methodology combined with a relational database as part of the implementation.

Booktwo of Object-oriented Knowledge

Corey Ladas' groundbreaking paper "ScrumBan" has captured the imagination of

the software development world. Scrum and agile methodologies have helped software development teams organize and become more efficient. Lean methods like kanban can extend these benefits. Kanban also provides a powerful mechanism to identify process improvement opportunities. This book covers some of the metrics and day-to-day management techniques that make continuous improvement an achievable outcome in the real world. ScrumBan the book provides a series of essays that give practitioners the background needed to create more robust practices combining the best of agile and lean.

Software Quality

This revision of Grady Booch's classic offers the first industry-wide standard for notation in developing large scale object-oriented systems. Laying the groundwork for the development of complex systems based on the object model, the author works in C++ to provide five fully-developed design examples, along with many smaller applications. Three of these capstone projects are new with this edition, including an inventory tracking system which implements a client server. The other four span problem domains as diverse as data acquisition for scientific tools, framework, artificial intelligence, and command and control. To measure progress, metrics in object development are suggested so that the developer knows how the project is going. In addition, the author demonstrates good and bad object designs and shows how to manage the trade-offs in complex systems.

Object Expo Europe Conference Proceedings

Object-oriented (OO) metrics are an integral part of object technology -- at the research level and in commercial software development projects. This book offers theoretical and empirical tips and facts for creating an OO complexity metrics (measurement) program, based on a review of existing research from the last several years. KEY TOPICS: Covers moving through object-oriented concepts as they related to managing the project lifecycle; the framework in which metrics exist; structural complexity metrics for traditional systems; OO product metrics; and current industrial applications. MARKET: For software developers, programmers, and managers.

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