

Programming Distributed Computing Systems A Foundational Approach

Right here, we have countless book **Programming Distributed Computing Systems A Foundational Approach** and collections to check out. We additionally pay for variant types and moreover type of the books to browse. The normal book, fiction, history, novel, scientific research, as skillfully as various other sorts of books are readily easy to get to here.

As this Programming Distributed Computing Systems A Foundational Approach , it ends going on monster one of the favored books Programming Distributed Computing Systems A Foundational Approach collections that we have. This is why you remain in the best website to see the incredible ebook to have.

Actors - Gul A. Agha 1986

Advancing the understanding of parallel computation.

Distributed Simulation - John A. Hamilton 2020-08-18

Simulation is a multi-disciplinary field, and significant simulation research is dispersed across multiple fields of study. Distributed computer systems, software design methods, and new simulation techniques offer synergistic multipliers when joined together in a distributed simulation. Systems of most interest to the simulation practitioner are often the most difficult to model and implement. Distributed Simulation brings together the many complex technologies for distributed simulation. There is strong emphasis on emerging simulation methodologies, including object-oriented, multilevel, and multi-resolution simulation. Finally, one concise text provides a strong foundation for the development of high fidelity simulations in heterogeneous distributed computing environments!

Parallel and Distributed Computation: Numerical Methods - Dimitri Bertsekas 2015-03-01

This highly acclaimed work, first published by Prentice Hall in 1989, is a comprehensive and theoretically sound treatment of parallel and distributed numerical methods. It focuses on algorithms that are naturally suited for massive parallelization, and it explores the fundamental convergence, rate of convergence, communication, and synchronization issues associated with such algorithms. This is an extensive book, which aside from its focus on parallel and distributed algorithms, contains a wealth of material on a broad variety of computation and optimization topics. It is an excellent supplement to several of our other books, including *Convex Optimization Algorithms* (Athena Scientific, 2015), *Nonlinear Programming* (Athena Scientific, 1999), *Dynamic Programming and Optimal Control* (Athena Scientific, 2012), *Neuro-Dynamic Programming* (Athena Scientific, 1996), and *Network Optimization* (Athena Scientific, 1998). The on-line edition of the book contains a 95-page solutions manual.

A Discipline of Multiprogramming - Jayadev Misra 2001-06-26

In this book, a programming model is developed that addresses the fundamental issues of 'large-scale programming'. The approach unifies several concepts from database theory, object-oriented programming and designs of reactive systems. The model and the associated theory has been christened "Seuss." The major goal of Seuss is to simplify multiprogramming. To this end, the concern of concurrent implementation is separated from the core program design problem. A program execution is understood as a single thread of control - sequential executions of actions that are chosen according to some scheduling policy. As a consequence, it is possible to reason about the properties of a program from its single execution thread.

Distributed Operating Systems - Doreen L. Galli 2000

Doreen Galli uses her considerable academic and professional experience to bring together the worlds of theory and practice providing leading edge solutions to tomorrow's challenges. "Distributed Operating Systems: Concepts and Practice" offers a good balance of real world examples and the underlying theory of distributed computing. The flexible design makes it usable for students, practitioners and corporate training. This book describes in detail each major aspect of distributed operating systems from a conceptual and practical viewpoint. The operating systems of Amoeba, Clouds, and Chorus(TM) (the base technology for JavaOS(TM)) are utilized as examples

throughout the text; while the technologies of Windows 2000(TM), CORBA(TM), DCOM(TM), NFS, LDAP, X.500, Kerberos, RSA(TM), DES, SSH, and NTP demonstrate real life solutions. A simple client/server application is included in the appendix to demonstrate key distributed computing programming concepts. This book proves invaluable as a course text or as a reference book for those who wish to update and enhance their knowledge base. A Companion Website provides supplemental information. A broad range of distributed computing issues and concepts: Kernels, IPC, memory management, object-based operating systems, distributed file systems (with NFS and X.500), transaction management, process management, distributed synchronization, and distributed security A major case study of Windows 2000 to demonstrate a real life commercial solution Detail Boxes contain in-depth examples such as complex algorithms Project-oriented exercises providing hands-on-experience Relevant sources including 'core' Web and ftp sites, as well as research papers Easy reference with complete list of acronyms and glossary to aid readability

Replication Techniques in Distributed Systems - Abdelsalam A. Helal 1996-08-31

Replication Techniques in Distributed Systems organizes and surveys the spectrum of replication protocols and systems that achieve high availability by replicating entities in failure-prone distributed computing environments. The entities discussed in this book vary from passive untyped data objects, to typed and complex objects, to processes and messages. Replication Techniques in Distributed Systems contains definitions and introductory material suitable for a beginner, theoretical foundations and algorithms, an annotated bibliography of commercial and experimental prototype systems, as well as short guides to recommended further readings in specialized subtopics. This book can be used as recommended or required reading in graduate courses in academia, as well as a handbook for designers and implementors of systems that must deal with replication issues in distributed systems.

Encyclopedia of Cloud Computing - San Murugesan 2016-08-01

The Encyclopedia of Cloud Computing provides IT professionals, educators, researchers and students with a compendium of cloud computing knowledge. Authored by a spectrum of subject matter experts in industry and academia, this unique publication, in a single volume, covers a wide range of cloud computing topics, including technological trends and developments, research opportunities, best practices, standards, and cloud adoption. Providing multiple perspectives, it also addresses questions that stakeholders might have in the context of development, operation, management, and use of clouds. Furthermore, it examines cloud computing's impact now and in the future. The encyclopedia presents 56 chapters logically organized into 10 sections. Each chapter covers a major topic/area with cross-references to other chapters and contains tables, illustrations, side-bars as appropriate. Furthermore, each chapter presents its summary at the beginning and backend material, references and additional resources for further information.

Intelligent Distributed Computing, Systems and Applications - Costin Badica 2008-08-25

This book represents the peer-reviewed proceedings of the Second International Symposium on Intelligent Distributed Computing – IDC 2008 held in Catania, Italy during September 18-19, 2008. The 35 contributions in this book address many topics related to intelligent and distributed computing,

systems and applications, including: adaptivity and learning; agents and multi-agent systems; argumentation; auctions; case-based reasoning; collaborative systems; data structures; distributed algorithms; formal modeling and verification; genetic and immune algorithms; grid computing; information extraction, annotation and integration; network and security protocols; mobile and ubiquitous computing; ontologies and metadata; P2P computing; planning; recommender systems; rules; semantic Web; services and processes; trust and social computing; virtual organizations; wireless networks; XML technologies.

Service-Oriented Computing - Xavier Franch 2014-10-10

This book constitutes the refereed conference proceedings of the 12th International Conference on Service-Oriented Computing, ICSOC 2014, held in Paris, France, in November 2014. The 25 full and 26 short papers presented were carefully reviewed and selected from 180 submissions. The papers are organized in topical sections on business process management; service composition and discovery; service design, description and evolution; cloud and business service management; ensuring composition properties; quality of service; semantic web services; service management; cloud service management; business service management; trust; service design and description.

Programming Distributed Computing Systems - Carlos A. Varela 2013-05-31

An introduction to fundamental theories of concurrent computation and associated programming languages for developing distributed and mobile computing systems. Starting from the premise that understanding the foundations of concurrent programming is key to developing distributed computing systems, this book first presents the fundamental theories of concurrent computing and then introduces the programming languages that help develop distributed computing systems at a high level of abstraction. The major theories of concurrent computation—including the π -calculus, the actor model, the join calculus, and mobile ambients—are explained with a focus on how they help design and reason about distributed and mobile computing systems. The book then presents programming languages that follow the theoretical models already described, including Pict, SALSA, and JoCaml. The parallel structure of the chapters in both part one (theory) and part two (practice) enable the reader not only to compare the different theories but also to see clearly how a programming language supports a theoretical model. The book is unique in bridging the gap between the theory and the practice of programming distributed computing systems. It can be used as a textbook for graduate and advanced undergraduate students in computer science or as a reference for researchers in the area of programming technology for distributed computing. By presenting theory first, the book allows readers to focus on the essential components of concurrency, distribution, and mobility without getting bogged down in syntactic details of specific programming languages. Once the theory is understood, the practical part of implementing a system in an actual programming language becomes much easier.

Grid and Distributed Computing - Dominik Slezak 2010-05-10

As future generation information technology (FGIT) becomes specialized and fragmented, it is easy to lose sight that many topics in FGIT have common threads and, because of this, advances in one discipline may be transmitted to others. Presentation of recent results obtained in different disciplines encourages this interchange for the advancement of FGIT as a whole. Of particular interest are hybrid solutions that combine ideas taken from multiple disciplines in order to achieve something more significant than the sum of the individual parts. Through such hybrid philosophy, a new principle can be discovered, which has the propensity to propagate throughout multifaceted disciplines. FGIT 2009 was the first mega-conference that attempted to follow the above idea of hybridization in FGIT in a form of multiple events related to particular disciplines of IT, conducted by separate scientific committees, but coordinated in order to expose the most important contributions. It included the following international conferences: Advanced Software Engineering and Its Applications (ASEA), Bio-Science and Bio-Technology (BSBT), Control and Automation (CA), Database Theory and Application (DTA), Disaster Recovery and Business Continuity (DRBC; published independently), Future Generation Communication and Networking (FGCN) that was combined with Advanced Communication and Networking (ACN), Grid and Distributed Computing

(GDC), Multimedia, Computer Graphics and Broadcasting (MulGraB), Security Technology (SecTech), Signal Processing, Image Processing and Pattern Recognition (SIP), and e-Service, Science and Technology (UNESST).

Foundations of Multithreaded, Parallel, and Distributed Programming -

Gregory R. Andrews 2000

Foundations of Multithreaded, Parallel, and Distributed Programming covers, and then applies, the core concepts and techniques needed for an introductory course in this subject. Its emphasis is on the practice and application of parallel systems, using real-world examples throughout. Greg Andrews teaches the fundamental concepts of multithreaded, parallel and distributed computing and relates them to the implementation and performance processes. He presents the appropriate breadth of topics and supports these discussions with an emphasis on performance. Features Emphasizes how to solve problems, with correctness the primary concern and performance an important, but secondary, concern Includes a number of case studies which cover such topics as pthreads, MPI, and OpenMP libraries, as well as programming languages like Java, Ada, high performance Fortran, Linda, Occam, and SR Provides examples using Java syntax and discusses how Java deals with monitors, sockets, and remote method invocation Covers current programming techniques such as semaphores, locks, barriers, monitors, message passing, and remote invocation Concrete examples are executed with complete programs, both shared and distributed Sample applications include scientific computing and distributed systems 0201357526B04062001

Interacting Processes - Nissim Francez 1996

In response to the industry's need for coordination, this book represents an approach to the design of coordinated, distributed programs, based on a high-level language, IP. This book appeals to theoretical computer scientists who are interested in the application of formal methods to distributed programs and software engineers who adopt an algorithmic approach when they develop software for distributed systems.

Distributed Network Systems - Weijia Jia 2004-12-15

Both authors have taught the course of "Distributed Systems" for many years in the respective schools. During the teaching, we feel strongly that "Distributed systems" have evolved from traditional "LAN" based distributed systems towards "Internet based" systems. Although there exist many excellent textbooks on this topic, because of the fast development of distributed systems and network programming/protocols, we have difficulty in finding an appropriate textbook for the course of "distributed systems" with orientation to the requirement of the undergraduate level study for today's distributed technology. Specifically, from up-to-date concepts, algorithms, and models to implementations for both distributed system designs and application programming. Thus the philosophy behind this book is to integrate the concepts, algorithm designs and implementations of distributed systems based on network programming. After using several materials of other textbooks and research books, we found that many texts treat the distributed systems with separation of concepts, algorithm design and network programming and it is very difficult for students to map the concepts of distributed systems to the algorithm design, prototyping and implementations. This book intends to enable readers, especially postgraduates and senior undergraduate level, to study up-to-date concepts, algorithms and network programming skills for building modern distributed systems. It enables students not only to master the concepts of distributed network system but also to readily use the material introduced into implementation practices.

Distributed Computer-Aided Engineering - Hojjat Adeli 1998-10-28

Networking of personal computers and workstations is becoming commonplace in academic and industrial environments. A cluster of workstations provides engineers with a familiar, cost-effective environment for high performance computing. However, workstations often have no dedicated link and communicate slowly on a local area network (LAN), such as the Ethernet. Thus, to effectively harness the parallel processing or distributed computing capabilities of workstations, new algorithms need to be developed with a higher computation-to-communication ratio. Distributed Computer-Aided Engineering presents distributed algorithms for three fundamental areas: finite element analysis, design optimization, and

visualization - providing a new direction in high performance structural engineering computing.

Foundations of Secure Computation - Friedrich L. Bauer 2000

The final quarter of the 20th century has seen the establishment of a global computational infrastructure. This and the advent of programming languages such as Java, supporting mobile distributed computing, has posed a significant challenge to computer sciences. The infrastructure can support commerce, medicine and government, but only if communications and computing can be secured against catastrophic failure and malicious interference.

Parallel and Distributed Computing - Claudia Leopold 2001

An all-inclusive survey of the fundamentals of parallel and distributed computing. The use of parallel and distributed computing has increased dramatically over the past few years, giving rise to a variety of projects, implementations, and buzzwords surrounding the subject. Although the areas of parallel and distributed computing have traditionally evolved separately, these models have overlapping goals and characteristics. Parallel and Distributed Computing surveys the models and paradigms in this converging area of parallel and distributed computing and considers the diverse approaches within a common text. Covering a comprehensive set of models and paradigms, the material also skims lightly over more specific details and serves as both an introduction and a survey. Novice readers will be able to quickly grasp a balanced overview with the review of central concepts, problems, and ideas, while the more experienced researcher will appreciate the specific comparisons between models, the coherency of the parallel and distributed computing field, and the discussion of less well-known proposals. Other topics covered include: * Data parallelism * Shared-memory programming * Message passing * Client/server computing * Code mobility * Coordination, object-oriented, high-level, and abstract models * And much more Parallel and Distributed Computing is a perfect tool for students and can be used as a foundation for parallel and distributed computing courses.

Application developers will find this book helpful to get an overview before choosing a particular programming style to study in depth, and researchers and programmers will appreciate the wealth of information concerning the various areas of parallel and distributed computing.

Distributed Algorithms - Wan Fokkink 2013-12-06

A comprehensive guide to distributed algorithms that emphasizes examples and exercises rather than mathematical argumentation.

Programming Distributed Computing Systems : a Foundational Approach - Carlos A. Varela 2013

Distributed Computing - Ajay D. Kshemkalyani 2011-03-03

Designing distributed computing systems is a complex process requiring a solid understanding of the design problems and the theoretical and practical aspects of their solutions. This comprehensive textbook covers the fundamental principles and models underlying the theory, algorithms and systems aspects of distributed computing. Broad and detailed coverage of the theory is balanced with practical systems-related issues such as mutual exclusion, deadlock detection, authentication, and failure recovery. Algorithms are carefully selected, lucidly presented, and described without complex proofs. Simple explanations and illustrations are used to elucidate the algorithms. Important emerging topics such as peer-to-peer networks and network security are also considered. With vital algorithms, numerous illustrations, examples and homework problems, this textbook is suitable for advanced undergraduate and graduate students of electrical and computer engineering and computer science. Practitioners in data networking and sensor networks will also find this a valuable resource. Additional resources are available online at www.cambridge.org/9780521876346.

Foundation of Software Science and Computation Structures - Jerzy Tiuryn 2003-06-26

ETAPS2000 was the third instance of the European Joint Conferences on Theory and Practice of Software. ETAPS is an annual federated conference that was established in 1998 by combining a number of existing and new conferences. This year it comprised five conferences (FOSSACS, FASE, ESOP, CC, TACAS), seven satellite workshops (CBS, CMCS, CoFI, GRATRA, INT), seven invited

lectures, a panel discussion, and ten tutorials. The events that comprise ETAPS address various aspects of the system - development process, including specification, design, implementation, analysis, and improvement. The languages, methodologies, and tools which support these activities are all well within its scope. The recent blends of theory and practice are represented, with an inclination towards theory with a practical motivation on one hand and soundly-based practice on the other. Many of the issues involved in software design apply to systems in general, including hardware systems, and the emphasis on software is not intended to be exclusive. ETAPS is a loose confederation in which each event retains its own identity, with a separate program committee and independent proceedings. Its format is open-ended, allowing it to grow and evolve as time goes by. Contributed talks and system demonstrations are in synchronized parallel sessions, with invited lectures in plenary sessions. Two of the invited lectures are reserved for "unifying" talks on topics of interest to the whole range of ETAPS attendees. *Concurrent and Distributed Computing in Java* - Vijay K. Garg 2005-01-28 Concurrent and Distributed Computing in Java addresses fundamental concepts in concurrent computing with Java examples. The book consists of two parts. The first part deals with techniques for programming in shared-memory based systems. The book covers concepts in Java such as threads, synchronized methods, waits, and notify to expose students to basic concepts for multi-threaded programming. It also includes algorithms for mutual exclusion, consensus, atomic objects, and wait-free data structures. The second part of the book deals with programming in a message-passing system. This part covers resource allocation problems, logical clocks, global property detection, leader election, message ordering, agreement algorithms, checkpointing, and message logging. Primarily a textbook for upper-level undergraduates and graduate students, this thorough treatment will also be of interest to professional programmers.

Distributed Computing Through Combinatorial Topology - Maurice Herlihy 2013-11-30

Distributed Computing Through Combinatorial Topology describes techniques for analyzing distributed algorithms based on award winning combinatorial topology research. The authors present a solid theoretical foundation relevant to many real systems reliant on parallelism with unpredictable delays, such as multicore microprocessors, wireless networks, distributed systems, and Internet protocols. Today, a new student or researcher must assemble a collection of scattered conference publications, which are typically terse and commonly use different notations and terminologies. This book provides a self-contained explanation of the mathematics to readers with computer science backgrounds, as well as explaining computer science concepts to readers with backgrounds in applied mathematics. The first section presents mathematical notions and models, including message passing and shared-memory systems, failures, and timing models. The next section presents core concepts in two chapters each: first, proving a simple result that lends itself to examples and pictures that will build up readers' intuition; then generalizing the concept to prove a more sophisticated result. The overall result weaves together and develops the basic concepts of the field, presenting them in a gradual and intuitively appealing way. The book's final section discusses advanced topics typically found in a graduate-level course for those who wish to explore further. Named a 2013 Notable Computer Book for Computing Methodologies by Computing Reviews Gathers knowledge otherwise spread across research and conference papers using consistent notations and a standard approach to facilitate understanding Presents unique insights applicable to multiple computing fields, including multicore microprocessors, wireless networks, distributed systems, and Internet protocols Synthesizes and distills material into a simple, unified presentation with examples, illustrations, and exercises

Reflections on Programming Systems - Liesbeth De Mol 2019-01-10

This book presents a historical and philosophical analysis of programming systems, intended as large computational systems like, for instance, operating systems, programmed to control processes. The introduction to the volume emphasizes the contemporary need of providing a foundational analysis of such systems, rooted in a broader historical and philosophical discussion. The

different chapters are grouped around three major themes. The first concerns the early history of large systems developed against the background of issues related to the growing semantic gap between hardware and code. The second revisits the fundamental issue of complexity of large systems, dealt with by the use of formal methods and the development of 'grand designs' like Unix. Finally, a third part considers several issues related to programming systems in the real world, including chapters on aesthetical, ethical and political issues. This book will interest researchers from a diversity of backgrounds. It will appeal to historians, philosophers, as well as logicians and computer scientists who want to engage with topics relevant to the history and philosophy of programming and more specifically the role of programming systems in the foundations of computing.

Distributed Systems - Sukumar Ghosh 2014-07-14

Distributed Systems: An Algorithmic Approach, Second Edition provides a balanced and straightforward treatment of the underlying theory and practical applications of distributed computing. As in the previous version, the language is kept as unobscured as possible—clarity is given priority over mathematical formalism. This easily digestible text: Features significant updates that mirror the phenomenal growth of distributed systems Explores new topics related to peer-to-peer and social networks Includes fresh exercises, examples, and case studies Supplying a solid understanding of the key principles of distributed computing and their relationship to real-world applications, Distributed Systems: An Algorithmic Approach, Second Edition makes both an ideal textbook and a handy professional reference.

Understanding Distributed Systems - Roberto Vitillo 2021

Learning to build distributed systems is hard, especially if they are large scale. It's not that there is a lack of information out there. You can find academic papers, engineering blogs, and even books on the subject. The problem is that the available information is spread out all over the place, and if you were to put it on a spectrum from theory to practice, you would find a lot of material at the two ends, but not much in the middle. That is why I decided to write a book to teach the fundamentals of distributed systems so that you don't have to spend countless hours scratching your head to understand how everything fits together. This is the guide I wished existed when I first started out, and it's based on my experience building large distributed systems that scale to millions of requests per second and billions of devices. If you develop the back-end of web or mobile applications (or would like to!), this book is for you. When building distributed systems, you need to be familiar with the network stack, data consistency models, scalability and reliability patterns, and much more. Although you can build applications without knowing any of that, you will end up spending hours debugging and re-designing their architecture, learning lessons that you could have acquired in a much faster and less painful way.

Distributed Systems - Sukumar Ghosh 2006-11-22

Most applications in distributed computing center around a set of common subproblems. Distributed Systems: An Algorithmic Approach presents the algorithmic issues and necessary background theory that are needed to properly understand these challenges. Achieving a balance between theory and practice, this book bridges the gap between theoreticians and practitioners. With a set of exercises featured in each chapter, the book begins with background information that contains various interprocess communication techniques and middleware services, followed by foundational topics that cover system models, correctness criteria, and proof techniques. The book also presents numerous important paradigms in distributed systems, including logical clocks, distributed snapshots, deadlock detection, termination detection, election, and several graph algorithms. The author then addresses failures and fault-tolerance techniques in diverse applications, such as consensus, transactions, group communication, replicated data management, and self-stabilization. He concludes with an exploration of real-world issues, including distributed discrete-event simulation and security, sensor networks, and peer-to-peer networks. By covering foundational matters of distributed systems and their relationships to real-world applications, Distributed Systems provides insight into common distributed computing subproblems,

Distributed Computer Control Systems 1986 - K.D. Muller 2014-05-23

This seventh IFAC workshop on distributed control systems (DCCS) discusses the ideas of real-time synchronization and data consistency in industry, with emphasis on the Manufacturing Automation Protocol (MAP). The volume also debates the gulf between the computer scientist's approach to language and the needs of the application programmer. In addition to treating relevant topics, each session has an introductory paper and a panel discussion, to give a complete picture of the progress and research in this computer field today.

ZooKeeper - Flavio Junqueira 2013-11-18

Building distributed applications is difficult enough without having to coordinate the actions that make them work. This practical guide shows how Apache ZooKeeper helps you manage distributed systems, so you can focus mainly on application logic. Even with ZooKeeper, implementing coordination tasks is not trivial, but this book provides good practices to give you a head start, and points out caveats that developers and administrators alike need to watch for along the way. In three separate sections, ZooKeeper contributors Flavio Junqueira and Benjamin Reed introduce the principles of distributed systems, provide ZooKeeper programming techniques, and include the information you need to administer this service. Learn how ZooKeeper solves common coordination tasks Explore the ZooKeeper API's Java and C implementations and how they differ Use methods to track and react to ZooKeeper state changes Handle failures of the network, application processes, and ZooKeeper itself Learn about ZooKeeper's trickier aspects dealing with concurrency, ordering, and configuration Use the Curator high-level interface for connection management Become familiar with ZooKeeper internals and administration tools

Distributed Systems - Sukumar Ghosh 2014-07-14

Distributed Systems: An Algorithmic Approach, Second Edition provides a balanced and straightforward treatment of the underlying theory and practical applications of distributed computing. As in the previous version, the language is kept as unobscured as possible—clarity is given priority over mathematical formalism. This easily digestible text: Features significant updates that mirror the phenomenal growth of distributed systems Explores new topics related to peer-to-peer and social networks Includes fresh exercises, examples, and case studies Supplying a solid understanding of the key principles of distributed computing and their relationship to real-world applications, Distributed Systems: An Algorithmic Approach, Second Edition makes both an ideal textbook and a handy professional reference.

Formal Methods at the Crossroads. From Panacea to Foundational Support - Bernhard K. Aichernig 2003-11-18

This volume is devoted to the 10th Anniversary Colloquium of UNU/IIST, the International Institute for Software Technology of the United Nations University, as well as to the memory of Armando Haeberer, who passed away while he was working on the preparation of this book in February 2003. The volume starts with a special paper by Tom Maibaum recollecting Armando Haeberer's life and work. The second part presents work done by members of UNU/IIST as well as a paper on the history of the institute. The subsequent topical sections present key contributions by leading researchers and thus assess the state of the art in software engineering and its engineering and scientific principles, from models to software, real-time systems, and verification. All in all, the book is a unique survey of the power and potential of formal methods in software engineering.

Nested Transactions - J. Eliot B. Moss 1985

This book shows how to program a system so that the software continues to work in the face of a variety of failures in parts of the system.

Distributed Systems Architecture - Arno Puder 2011-04-18

Middleware is the bridge that connects distributed applications across different physical locations, with different hardware platforms, network technologies, operating systems, and programming languages. This book describes middleware from two different perspectives: from the viewpoint of the systems programmer and from the viewpoint of the applications programmer. It focuses on the use of open source solutions for creating middleware and the tools for developing distributed applications. The design principles presented are universal and apply to all middleware platforms, including CORBA and Web Services. The authors have created an open-source implementation of

CORBA, called MICO, which is freely available on the web. MICO is one of the most successful of all open source projects and is widely used by demanding companies and institutions, and has also been adopted by many in the Linux community. * Provides a comprehensive look at the architecture and design of middleware the bridge that connects distributed software applications * Includes a complete, commercial-quality open source middleware system written in C++ * Describes the theory of the middleware standard CORBA as well as how to implement a design using open source techniques

Systems Programming - Richard Anthony 2015-02-25

Systems Programming: Designing and Developing Distributed Applications explains how the development of distributed applications depends on a foundational understanding of the relationship among operating systems, networking, distributed systems, and programming. Uniquely organized around four viewpoints (process, communication, resource, and architecture), the fundamental and essential characteristics of distributed systems are explored in ways which cut across the various traditional subject area boundaries. The structures, configurations and behaviours of distributed systems are all examined, allowing readers to explore concepts from different perspectives, and to understand systems in depth, both from the component level and holistically. Explains key ideas from the ground up, in a self-contained style, with material carefully sequenced to make it easy to absorb and follow. Features a detailed case study that is designed to serve as a common point of reference and to provide continuity across the different technical chapters. Includes a 'putting it all together' chapter that looks at interesting distributed systems applications across their entire life-cycle from requirements analysis and design specifications to fully working applications with full source code. Ancillary materials include problems and solutions, programming exercises, simulation experiments, and a wide range of fully working sample applications with complete source code developed in C++, C# and Java. Special editions of the author's established 'workbenches' teaching and learning tools suite are included. These tools have been specifically designed to facilitate practical experimentation and simulation of complex and dynamic aspects of systems.

Designing Distributed Systems - Brendan Burns 2018-02-20

Without established design patterns to guide them, developers have had to build distributed systems from scratch, and most of these systems are very unique indeed. Today, the increasing use of containers has paved the way for core distributed system patterns and reusable containerized components. This practical guide presents a collection of repeatable, generic patterns to help make the development of reliable distributed systems far more approachable and efficient. Author Brendan Burns—Director of Engineering at Microsoft Azure—demonstrates how you can adapt existing software design patterns for designing and building reliable distributed applications. Systems engineers and application developers will learn how these long-established patterns provide a common language and framework for dramatically increasing the quality of your system. Understand how patterns and reusable components enable the rapid development of reliable distributed systems Use the side-car,

adapter, and ambassador patterns to split your application into a group of containers on a single machine Explore loosely coupled multi-node distributed patterns for replication, scaling, and communication between the components Learn distributed system patterns for large-scale batch data processing covering work-queues, event-based processing, and coordinated workflows Fundamental Approaches to Software Engineering - Luciano Baresi 2006-03-16

This book constitutes the refereed proceedings of the 9th International Conference on Fundamental Approaches to Software Engineering, FASE 2006, held in Vienna, Austria in March 2006 as part of ETAPS. The 27 revised full papers, two tool papers presented together with two invited papers were carefully reviewed and selected from 166 submissions. The papers are organized in topical sections.

Leveraging Applications of Formal Methods, Verification and Validation: Engineering Principles - Tiziana Margaria 2020-10-26

The three-volume set LNCS 12476 - 12478 constitutes the refereed proceedings of the 9th International Symposium on Leveraging Applications of Formal Methods, ISoLA 2020, which was planned to take place during October 20–30, 2020, on Rhodes, Greece. The event itself was postponed to 2021 due to the COVID-19 pandemic. The papers presented were carefully reviewed and selected for inclusion in the proceedings. Each volume focusses on an individual topic with topical section headings within the volume: Part I, Verification Principles: Modularity and (De-)Composition in Verification; X-by-Construction: Correctness meets Probability; 30 Years of Statistical Model Checking; Verification and Validation of Concurrent and Distributed Systems. Part II, Engineering Principles: Automating Software Re-Engineering; Rigorous Engineering of Collective Adaptive Systems. Part III, Applications: Reliable Smart Contracts: State-of-the-art, Applications, Challenges and Future Directions; Automated Verification of Embedded Control Software; Formal methods for DIStributed COmputing in future RAILway systems.

Concurrent Systems - Jean Bacon 1998

This is a textbook on concurrent programming which serves to integrate operating systems and database concepts, and provides a foundation for later study in these areas.

Principles of Concurrent and Distributed Programming - M. Ben-Ari 1990

Principles of Concurrent and Distributed Programming provides an introduction to concurrent programming focusing on general principles and not on specific systems.

Fundamental Approaches to Software Engineering - Esther Guerra 2021-04-20

This open access book constitutes the proceedings of the 24th International Conference on Fundamental Approaches to Software Engineering, FASE 2021, which took place during March 27–April 1, 2021, and was held as part of the Joint Conferences on Theory and Practice of Software, ETAPS 2021. The conference was planned to take place in Luxembourg but changed to an online format due to the COVID-19 pandemic. The 16 full papers presented in this volume were carefully reviewed and selected from 52 submissions. The book also contains 4 Test-Comp contributions.