Introduction To Reliable And Secure Distributed Programming

If you ally compulsion such a referred **Introduction To Reliable And Secure Distributed Programming** book that will offer you worth, get the certainly best seller from us currently from several preferred authors. If you want to comical books, lots of novels, tale, jokes, and more fictions collections are after that launched, from best seller to one of the most current released.

You may not be perplexed to enjoy every books collections Introduction To Reliable And Secure Distributed Programming that we will completely offer. It is not on the order of the costs. Its about what you need currently. This Introduction To Reliable And Secure Distributed Programming , as one of the most working sellers here will unquestionably be among the best options to review.

Programming Distributed Computing Systems - Carlos A. Varela 2013 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.

SCION: A Secure Internet Architecture - Adrian Perrig 2017-10-13 This book describes the essential components of the SCION secure Internet architecture, the first architecture designed foremost for strong security and high availability. Among its core features, SCION also provides route control, explicit trust information, multipath communication, scalable quality-of-service guarantees, and efficient forwarding. The book includes functional specifications of the network elements, communication protocols among these elements, data structures, and configuration files. In particular, the book offers a specification of a working prototype. The authors provide a comprehensive description of the main design features for achieving a secure Internet architecture. They facilitate the reader throughout, structuring the book so that the technical detail gradually increases, and supporting the text with a glossary, an index, a list of abbreviations, answers to frequently asked questions, and special highlighting for examples and for sections that explain important research, engineering, and deployment features. The book is suitable for researchers, practitioners, and graduate students who are interested in network security.

Replication - Bernadette Charron-Bost 2010-03-02

Consistency models for replicated data /Alan D. Fekete and Krithi Ramamritham --Replication techniques for availability /Robbert van Renesse and Rachid Guerraoui --Modular approach to replication for availability /Fernando Pedone and André Schiper --Stumbling over consensus research: misunderstandings and issues /Marcos K. Aguilera --Replicating for performance: case studies /Maarten van Steen and Guillaume Pierre --A history of the virtual synchrony replication model /Ken Birman --From viewstamped replication to byzantine fault tolerance /Barbara Liskov --Implementing trustworthy services using replicated state machines /Fred B. Schneider and Lidong Zhou --State machine replication with Byzantine faults /Christian Cachin --Selected results from the latest decade of quorum systems research /Michael G. Merideth and Michael K. Reiter --From object replication to database replication /Fernando Pedone and André Schiper --Database replication: a tutorial /Dettina Kemme, Ricardo Jiménez-Peris, Marta Patiño-Martínez, and Gustavo Alonso --Practical database replication /Alfrânio Correia Jr. ... [et al.].

Distributed Programming with Ruby - Mark Bates 2009-11-05

Complete, Hands-On Guide to Building Advanced Distributed Applications with Ruby Distributed programming techniques make applications easier to scale, develop, and deploy—especially in emerging cloud computing environments. Now, one of the Ruby community's leading experts has written the first definitive guide to distributed programming with Ruby. Mark Bates begins with a simple distributed application, and then walks through an increasingly complex series of examples, demonstrating solutions to the most common distributed programming problems. Bates presents the industry's most useful coverage of Ruby's standard distributed programming libraries, DRb and Rinda. Next, he introduces powerful third-party tools, frameworks, and libraries designed to simplify Ruby distributed programming, including his own Distribunaut. If you're an experienced Ruby programmer or architect, this hands-on tutorial and practical reference will help you meet any distributed programming challenge, no matter how complex. Coverage includes Writing robust, secure, and interactive applications using DRb—and managing its drawbacks Using Rinda to build applications with improved flexibility, fault tolerance, and service discovery Simplifying DRb service management with RingyDingy Utilizing Starfish to facilitate communication between distributed programs and to write MapReduce functions for processin large data sets Using Politics to customize the processes running on individual server instances in a cloud computing environment Providing reliable distributed queuing with the low-overhead Starling messaging server Implementing comprehensive enterprise messaging with RabbitMQ and Advanced Message Queuing Protocol (AMQP) Offloading heavyweight tasks with BackgrounDRb and DelayedJob

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.

Building Secure and Reliable Systems - Heather Adkins 2020-03-16
Can a system be considered truly reliable if it isn't fundamentally secure? Or can it be considered secure if it's unreliable? Security is crucial to the design and operation of scalable systems in production, as it plays an important part in product quality, performance, and availability. In this book, experts from Google share best practices to help your organization design scalable and reliable systems that are fundamentally secure. Two previous O'Reilly books from Google—Site Reliability Engineering and The Site Reliability Workbook—demonstrated how and why a commitment to the entire service lifecycle enables organizations to successfully build, deploy, monitor, and maintain software systems. In this latest guide, the authors offer insights into system design, implementation, and maintenance from practitioners who specialize in security and reliability. They also discuss how building and adopting their recommended best practices requires a culture that's supportive of such change. You'll learn about secure and reliable systems through: Design strategies Recommendations for coding, testing, and debugging practices Strategies to prepare for, respond to, and recover from incidents Cultural best practices that help teams across your organization collaborate effectively

Distributed Network Systems - Weijia Jia 2006-06-14
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 - 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 Systems - Andrew S. Tanenbaum 2016-02-26
This second edition of Distributed Systems, Principles & Paradigms, covers the principles, advanced concepts, and technologies of distributed systems in detail, including: communication, replication, fault tolerance, and security. Intended for use in a senior/graduate level distributed systems course or by professionals, this text systematically shows how distributed systems are designed and implemented in real systems.

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

Reliable Distributed Systems - Kenneth Birman 2006-07-02
Explains fault tolerance in clear terms, with concrete examples drawn from real-world settings Highly practical focus aimed at building "mission-critical" networked applications that remain secure

Distributed Services with Go - Travis Jeffery 2021-03-11
This is the book for Gophers who want to learn how to build distributed systems. You know the basics of Go and are eager to put your knowledge to work. Build distributed services that are highly available, resilient, and scalable. This book is just what you need to apply Go to real-world situations. Level up your engineering skills today. Take your Go skills to the next level by learning how to design, develop, and deploy a distributed service. Start from the bare essentials of storage handling, then work your way through networking a client and server, and finally to distributing server instances, deployment, and testing. All this will make coding in your day job or side projects easier, faster, and more fun. Create your own distributed services and contribute to open source projects. Build networked, secure clients and servers with gRPC. Gain insights into your systems and debug issues with observable services instrumented with metrics, logs, and traces. Operate your own Certificate Authority to authenticate internal web services with TLS. Automatically handle when nodes are added or removed to your cluster with service discovery. Coordinate distributed systems with replicated state machines powered by the Raft consensus algorithm. Lay out your applications and libraries to be modular and easy to maintain. Write CLIs to configure and run your applications. Run your distributed system locally and deploy to the cloud with Kubernetes. Test and benchmark your applications to ensure they're correct and fast. Dive into writing Go and join the hundreds of thousands who are using it to build software for the real world. What You Need: Go 1.13+ and Kubernetes 1.16+

Introduction to Reliable and Secure Distributed Programming - Christian Cachin 2011-02-11
In modern computing a program is usually distributed among several processes. The fundamental challenge when developing reliable and secure distributed programs is to support the cooperation of processes required to execute a common task, even when some of these processes fail. Failures may range from crashes to adversarial attacks by malicious processes. Cachin, Guerraoui, and Rodrigues present an introductory description of fundamental distributed programming abstractions together with algorithms to implement them in distributed systems, where processes are subject to crashes and malicious attacks. The authors follow an incremental approach by first introducing basic abstractions in simple distributed environments, before moving to more sophisticated abstractions and more challenging environments. Each core chapter is devoted to one topic, covering reliable broadcast, shared memory, consensus, and extensions of consensus. For every topic, many exercises and their solutions enhance the understanding This book represents the second edition of "Introduction to Reliable Distributed Programming". Its scope has been extended to include security against malicious actions by non-cooperating processes. This important domain has become widely known under the name "Byzantine fault-tolerance".

Network Programming with Go - Adam Woodbeck 2021-03-30
Network Programming with Go teaches you how to write clean, secure network software with the programming language designed to make it seem easy. Go combines the best parts of many other programming languages. It’s fast, scalable, and designed for high-performance networking and multiprocessing—in other words, it’s perfect for network programming. Network Programming with Go is for developers ready to start leveraging Go’s ease of use for writing secure, readable, production-ready network code. Early chapters establish a foundation of networking and traffic-routing know-how upon which the rest of the book builds. You’ll put that knowledge to use as author Adam Woodbeck guides you through writing programs that communicate using TCP, UDP, Unix sockets, and other features that ensure reliable data transmission. As you progress, you’ll explore higher-level network protocols like HTTP and HTTP/2, then build applications that securely interact with servers,

clients, and APIs over a network using TLS. In addition, Woodbeck shows you how to create a simple messaging protocol, develop tools for monitoring network traffic, craft a custom web server, and implement best practices for interacting with cloud providers using their SDKs. Along the way, you'll learn: • IP basics for writing effective network programs, such as IPv4 and IPv6 multicasting, ports, and network address translation • How to use handlers, middleware, and multiplexers to build capable HTTP-based applications with minimal code • The OSI and TCP/IP models for layered data architectures • Methods for reading data from/writing data to a network connection, like the type-length-value encoding scheme • Tools for incorporating authentication and encryption into your applications using TLS, like mutual authentication • How to serialize data for storage or transmission in Go-friendly formats like JSON, Gob, XML, and protocol buffers • How to Leverage Go's code generation support to efficiently communicate with gRPC-based network services So get ready to take advantage of Go's built-in concurrency, rapid compiling, and rich standard library. Because when it comes to writing robust network programs, it's Go time.

Web Database Applications with PHP and MySQL - Hugh E. Williams 2002

Combines language tutorials with application design advice to cover the PHP server-side scripting language and the MySQL database engine.

Distributed Computing with Python - Francesco Pierfederici 2016-04-12

Harness the power of multiple computers using Python through this fast-paced informative guide About This Book You'll learn to write data processing programs in Python that are highly available, reliable, and fault tolerant Make use of Amazon Web Services along with Python to establish a powerful remote computation system Train Python to handle data-intensive and resource hungry applications Who This Book Is For This book is for Python developers who have developed Python programs for data processing and now want to learn how to write fast, efficient programs that perform CPU-intensive data processing tasks. What You Will Learn Get an introduction to parallel and distributed computing See synchronous and asynchronous programming Explore parallelism in Python Distributed application with Celery Python in the Cloud Python on an HPC cluster Test and debug distributed applications In Detail CPU-intensive data processing tasks have become crucial considering the complexity of the various big data applications that are used today. Reducing the CPU utilization per process is very important to improve the overall speed of applications. This book will teach you how to perform parallel execution of computations by distributing them across multiple processors in a single machine, thus improving the overall performance of a big data processing task. We will cover synchronous and asynchronous models, shared memory and file systems, communication between various processes, synchronization, and more. Style and Approach This example based, step-by-step guide will show you how to make the best of your hardware configuration using Python for distributing applications.

Database Internals - Alex Petrov 2019-09-13

When it comes to choosing, using, and maintaining a database, understanding its internals is essential. But with so many distributed databases and tools available today, it's often difficult to understand what each one offers and how they differ. With this practical guide, Alex Petrov guides developers through the concepts behind modern database and storage engine internals. Throughout the book, you'll explore relevant material gleaned from numerous books, papers, blog posts, and the source code of several open source databases. These resources are listed at the end of parts one and two. You'll discover that the most significant distinctions among many modern databases reside in subsystems that determine how storage is organized and how data is distributed. This book examines: Storage engines: Explore storage classification and taxonomy, and dive into B-Tree-based and immutable Log Structured storage engines, with differences and use-cases for each Storage building blocks: Learn how database files are organized to build efficient storage, using auxiliary data structures such as Page Cache, Buffer Pool and Write-Ahead Log Distributed systems: Learn step-by-step how nodes and processes connect and build complex communication patterns Database clusters: Which consistency models are commonly used by modern databases and how distributed storage systems achieve consistency

Research Anthology on Architectures, Frameworks, and Integration Strategies for Distributed and Cloud Computing - Management Association, Information Resources 2021-01-25

Distributed systems intertwine with our everyday lives. The benefits and current shortcomings of the underpinning technologies are experienced by a wide range of people and their smart devices. With the rise of large-scale IoT and similar distributed systems, cloud bursting technologies, and partial outsourcing solutions, private entities are encouraged to increase their efficiency and offer unparalleled availability and reliability to their users. The Research Anthology on Architectures, Frameworks, and Integration Strategies for Distributed and Cloud Computing is a vital reference source that provides valuable insight into current and emergent research occurring within the field of distributed computing. It also presents architectures and service frameworks to achieve highly integrated distributed systems and solutions to integration and efficient management challenges faced by current and future distributed systems. Highlighting a range of topics such as data sharing, wireless sensor networks, and scalability, this multi-volume book is ideally designed for system administrators, integrators, designers, developers, researchers, academicians, and students.

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.

Site Reliability Engineering - Niall Richard Murphy 2016-03-23

The overwhelming majority of a software system’s lifespan is spent in use, not in design or implementation. So, why does conventional wisdom insist that software engineers focus primarily on the design and development of large-scale computing systems? In this collection of essays and articles, key members of Google’s Site Reliability Team explain how and why their commitment to the entire lifecycle has enabled the company to successfully build, deploy, monitor, and maintain some of the largest software systems in the world. You’ll learn the principles and practices that enable Google engineers to make systems more scalable, reliable, and efficient—lessons directly applicable to your organization. This book is divided into four sections: Introduction—Learn what site reliability engineering is and why it differs from conventional IT industry practices Principles—Examine the patterns, behaviors, and areas of concern that influence the work of a site reliability engineer (SRE) Practices—Understand the theory and practice of an SRE’s day-to-day work: building and operating large distributed computing systems Management—Explore Google's best practices for training, communication, and meetings that your organization can use

Cyber Security in Parallel and Distributed Computing - Dac-Nhuong Le 2019-04-16

The book contains several new concepts, techniques, applications and case studies for cyber securities in parallel and distributed computing The main objective of this book is to explore the concept of cybersecurity in parallel and distributed computing along with recent research developments in the field. Also included are various real-time/offline applications and case studies in the fields of engineering and computer science and the modern tools and technologies used. Information concerning various topics relating to cybersecurity technologies is organized within the sixteen chapters of this book. Some of the important topics covered include: Research and solutions for the problem of hidden image detection Security aspects of data mining and possible solution techniques A comparative analysis of various methods used in ecommerce security and how to perform secure payment transactions in an efficient manner Blockchain technology and how it is crucial to the security industry Security for the Internet of Things Security issues and challenges in distributed computing security such as heterogeneous computing, cloud computing, fog computing, etc. Demonstrates the administration task issue in unified cloud situations as a multi-target enhancement issue in light of security Explores the concepts of cybercrime and cybersecurity and presents the statistical impact it is having on organizations Security policies and mechanisms, various categories of attacks (e.g., denial-of-service), global security architecture, along with distribution of security mechanisms Security issues in the healthcare sector with existing solutions and emerging threats.

Network Security with OpenSSL - John Viega 2002-06-17

Most applications these days are at least somewhat network aware, but how do you protect those applications against common network security threats? Many developers are turning to OpenSSL, an open source version of SSL/TLS, which is the most widely used protocol for secure network communications.The OpenSSL library is seeing widespread adoption for web sites that require cryptographic functions to protect a broad range of sensitive information, such as credit card numbers and other financial transactions. The library is the only free, full-featured SSL implementation for C and C++, and it can be used programmatically or from the command line to secure most TCP-based network protocols.Network Security with OpenSSL enables developers to use this protocol much more effectively. Traditionally, getting something simple done in OpenSSL could easily take weeks. This concise book gives you the guidance you need to avoid pitfalls, while allowing you to take advantage of the library?s advanced features. And, instead of bogging you down in the technical details of how SSL works under the hood, this book provides only the information that is necessary to use OpenSSL safely and effectively. In step-by-step fashion, the book details the challenges in securing network communications, and shows you how to use OpenSSL tools to best meet those challenges.As a system or network administrator, you will benefit from the thorough treatment of the OpenSSL command-line interface, as well as from step-by-step directions for obtaining certificates and setting up your own certification authority. As a developer, you will further benefit from the in-depth discussions and examples of how to use OpenSSL in your own programs. Although OpenSSL is written in C, information on how to use OpenSSL with Perl, Python and PHP is also included.OpenSSL may well answer your need to protect sensitive data. If that?s the case, Network Security with OpenSSL is the only guide available on the subject.

Introduction to Reliable and Secure Distributed Programming - Christian Cachin 2014-10-14

In modern computing a program is usually distributed among several processes. The fundamental challenge when developing reliable and secure distributed programs is to support the cooperation of processes required to execute a common task, even when some of these processes fail. Failures may range from crashes to adversarial attacks by malicious processes. Cachin, Guerraoui, and Rodrigues present an introductory description of fundamental distributed programming abstractions together with algorithms to implement them in distributed systems, where processes are subject to crashes and malicious attacks. The authors follow an incremental approach by first introducing basic abstractions in simple distributed environments, before moving to more sophisticated abstractions and more challenging environments. Each core chapter is devoted to one topic, covering reliable broadcast, shared memory, consensus, and extensions of consensus. For every topic, many exercises and their solutions enhance the understanding This book represents the second edition of "Introduction to Reliable Distributed Programming". Its scope has been extended to include security against malicious actions by non-cooperating processes. This important domain has become widely known under the name "Byzantine fault-tolerance".

Security Engineering - Ross Anderson 2020-12-22

Now that there’s software in everything, how can you make anything secure? Understand how to engineer dependable systems with this newly updated classic In Security Engineering: A Guide to Building Dependable

Distributed Systems, Third Edition Cambridge University professor Ross Anderson updates his classic textbook and teaches readers how to design, implement, and test systems to withstand both error and attack. This book became a best-seller in 2001 and helped establish the discipline of security engineering. By the second edition in 2008, underground dark markets had let the bad guys specialize and scale up; attacks were increasingly on users rather than on technology. The book repeated its success by showing how security engineers can focus on usability. Now the third edition brings it up to date for 2020. As people now go online from phones more than laptops, most servers are in the cloud, online advertising drives the Internet and social networks have taken over much human interaction, many patterns of crime and abuse are the same, but the methods have evolved. Ross Anderson explores what security engineering means in 2020, including: How the basic elements of cryptography, protocols, and access control translate to the new world of phones, cloud services, social media and the Internet of Things Who the attackers are – from nation states and business competitors through criminal gangs to stalkers and playground bullies What they do – from phishing and carding through SIM swapping and software exploits to DDoS and fake news Security psychology, from privacy through ease-of-use to deception The economics of security and dependability – why companies build vulnerable systems and governments look the other way How dozens of industries went online – well or badly How to manage security and safety engineering in a world of agile development – from reliability engineering to DevSecOps The third edition of Security Engineering ends with a grand challenge: sustainable security. As we build ever more software and connectivity into safety-critical durable goods like cars and medical devices, how do we design systems we can maintain and defend for decades? Or will everything in the world need monthly software upgrades, and become unsafe once they stop?

Distributed Systems - Maarten van Steen 2017-02

For this third edition of -Distributed Systems, - the material has been thoroughly revised and extended, integrating principles and paradigms into nine chapters: 1. Introduction 2. Architectures 3. Processes 4. Communication 5. Naming 6. Coordination 7. Replication 8. Fault tolerance 9. Security A separation has been made between basic material and more specific subjects. The latter have been organized into boxed sections, which may be skipped on first reading. To assist in understanding the more algorithmic parts, example programs in Python have been included. The examples in the book leave out many details for readability, but the complete code is available through the book's Website, hosted at www.distributed-systems.net. A personalized digital copy of the book is available for free, as well as a printed version through Amazon.com.

Distributed Computing in Java 9 - Raja Malleswara Rao Pattamsetti 2017-06-30

Explore the power of distributed computing to write concurrent, scalable applications in Java About This Book Make the best of Java 9 features to write succinct code Handle large amounts of data using HPC Make use of AWS and Google App Engine along with Java to establish a powerful remote computation system Who This Book Is For This book is for basic to intermediate level Java developers who is aware of object-oriented programming and Java basic concepts. What You Will Learn Understand the basic concepts of parallel and distributed computing/programming Achieve performance improvement using parallel processing, multithreading, concurrency, memory sharing, and hpc cluster computing Get an in-depth understanding of Enterprise Messaging concepts with Java Messaging Service and Web Services in the context of Enterprise Integration Patterns Work with Distributed Database technologies Understand how to develop and deploy a distributed application on different cloud platforms including Amazon Web Service and Docker CaaS Concepts Explore big data technologies Effectively test and debug distributed systems Gain thorough knowledge of security standards for distributed applications including two-way Secure Socket Layer In Detail Distributed computing is the concept with which a bigger computation process is accomplished by splitting it into multiple smaller logical activities and performed by diverse systems, resulting in maximized performance in lower infrastructure investment. This book will teach you how to improve the performance of traditional applications through the usage of parallelism and optimized resource utilization in Java 9. After a brief introduction to the fundamentals of distributed and parallel computing, the book moves on to explain different ways of communicating with remote systems/objects in a distributed architecture. You will learn about asynchronous messaging with enterprise integration and related patterns, and how to handle large amount of data using HPC and implement distributed computing for databases. Moving on, it explains how to deploy distributed applications on different cloud platforms and self-contained application development. You will also learn about big data technologies and understand how they contribute to distributed computing. The book concludes with the detailed coverage of testing, debugging, troubleshooting, and security aspects of distributed applications so the programs you build are robust, efficient, and secure. Style and approach This is a step-by-step practical guide with real-world examples.

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.

Introduction to Distributed Algorithms - Gerard Tel 2000-09-28

Introduction : distributed systems - The model - Communication protocols - Routing algorithms - Deadlock-free packet switching - Wave and traversal algorithms - Election algorithms - Termination detection - Anonymous networks - Snapshots - Sense of direction and orientation - Synchrony in networks - Fault tolerance in distributed systems - Fault tolerance in asynchronous systems - Fault tolerance in synchronous systems - Failure detection - Stabilization.

Erlang Programming - Francesco Cesarini 2009-06-11

This book is an in-depth introduction to Erlang, a programming language ideal for any situation where concurrency, fault tolerance, and fast response is essential. Erlang is gaining widespread adoption with the advent of multi-core processors and their new scalable approach to concurrency. With this guide you'll learn how to write complex concurrent programs in Erlang, regardless of your programming background or experience. Written by leaders of the international Erlang community -- and based on their training material -- Erlang Programming focuses on the language's syntax and semantics, and explains pattern matching, proper lists, recursion, debugging, networking, and concurrency. This book helps you: Understand the strengths of Erlang and why its designers included specific features Learn the concepts behind concurrency and Erlang's way of handling it Write efficient Erlang programs while keeping code neat and readable Discover how Erlang fills the requirements for distributed systems Add simple graphical user interfaces with little effort Learn Erlang's tracing mechanisms for debugging concurrent and distributed systems Use the built-in Mnesia database and other table storage features Erlang Programming provides exercises at the end of each chapter and simple examples throughout the book.

Distributed and Cloud Computing - Kai Hwang 2013-12-18

Distributed and Cloud Computing: From Parallel Processing to the

Internet of Things offers complete coverage of modern distributed computing technology including clusters, the grid, service-oriented architecture, massively parallel processors, peer-to-peer networking, and cloud computing. It is the first modern, up-to-date distributed systems textbook; it explains how to create high-performance, scalable, reliable systems, exposing the design principles, architecture, and innovative applications of parallel, distributed, and cloud computing systems. Topics covered by this book include: facilitating management, debugging, migration, and disaster recovery through virtualization; clustered systems for research or ecommerce applications; designing systems as web services; and social networking systems using peer-to-peer computing. The principles of cloud computing are discussed using examples from open-source and commercial applications, along with case studies from the leading distributed computing vendors such as Amazon, Microsoft, and Google. Each chapter includes exercises and further reading, with lecture slides and more available online. This book will be ideal for students taking a distributed systems or distributed computing class, as well as for professional system designers and engineers looking for a reference to the latest distributed technologies including cloud, P2P and grid computing. Complete coverage of modern distributed computing technology including clusters, the grid, service-oriented architecture, massively parallel processors, peer-to-peer networking, and cloud computing Includes case studies from the leading distributed computing vendors: Amazon, Microsoft, Google, and more Explains how to use virtualization to facilitate management, debugging, migration, and disaster recovery Designed for undergraduate or graduate students taking a distributed systems course—each chapter includes exercises and further reading, with lecture slides and more available online

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

Principles of Computer System Design - Jerome H. Saltzer 2009-05-21

Principles of Computer System Design is the first textbook to take a principles-based approach to the computer system design. It identifies, examines, and illustrates fundamental concepts in computer system design that are common across operating systems, networks, database systems, distributed systems, programming languages, software engineering, security, fault tolerance, and architecture. Through carefully analyzed case studies from each of these disciplines, it demonstrates how to apply these concepts to tackle practical system design problems. To support the focus on design, the text identifies and explains abstractions that have proven successful in practice such as remote procedure call, client/service organization, file systems, data integrity, consistency, and authenticated messages. Most computer systems are built using a handful of such abstractions. The text describes how these abstractions are implemented, demonstrates how they are used in different systems, and prepares the reader to apply them in future designs. The book is recommended for junior and senior undergraduate students in Operating Systems, Distributed Systems, Distributed Operating Systems and/or Computer Systems Design courses; and professional computer systems designers. Features: Concepts of computer system design guided by fundamental principles. Cross-cutting approach that identifies abstractions common to networking, operating systems, transaction systems, distributed systems, architecture, and software engineering. Case studies that make the abstractions real: naming (DNS and the URL); file systems (the UNIX file system); clients and services (NFS); virtualization (virtual machines); scheduling (disk arms); security (TLS). Numerous pseudocode fragments that provide concrete examples of abstract concepts. Extensive support. The authors and MIT OpenCourseWare provide on-line, free of charge, open educational resources, including additional chapters, course syllabi, board layouts and slides, lecture videos, and an archive of lecture schedules, class assignments, and design projects.

Introduction to Parallel Computing - Roman Trobec 2018-09-27
Advancements in microprocessor architecture, interconnection technology, and software development have fueled rapid growth in parallel and distributed computing. However, this development is only of practical benefit if it is accompanied by progress in the design, analysis and programming of parallel algorithms. This concise textbook provides, in one place, three mainstream parallelization approaches, Open MPP, MPI and OpenCL, for multicore computers, interconnected computers and graphical processing units. An overview of practical parallel computing and principles will enable the reader to design efficient parallel programs for solving various computational problems on state-of-the-art personal computers and computing clusters. Topics covered range from parallel algorithms, programming tools, OpenMP, MPI and OpenCL, followed by experimental measurements of parallel programs' run-times, and by engineering analysis of obtained results for improved parallel execution performances. Many examples and exercises support the exposition.

Fault-Tolerant Message-Passing Distributed Systems - Michel Raynal 2018-09-08
This book presents the most important fault-tolerant distributed programming abstractions and their associated distributed algorithms, in particular in terms of reliable communication and agreement, which lie at the heart of nearly all distributed applications. These programming abstractions, distributed objects or services, allow software designers and programmers to cope with asynchrony and the most important types of failures such as process crashes, message losses, and malicious behaviors of computing entities, widely known under the term "Byzantine fault-tolerance". The author introduces these notions in an incremental manner, starting from a clear specification, followed by algorithms which are first described intuitively and then proved correct. The book also presents impossibility results in classic distributed computing models, along with strategies, mainly failure detectors and randomization, that allow us to enrich these models. In this sense, the book constitutes an introduction to the science of distributed computing, with applications in all domains of distributed systems, such as cloud computing and blockchains. Each chapter comes with exercises and bibliographic notes to help the reader approach, understand, and master the fascinating field of fault-tolerant distributed computing.

Blockchain for Distributed Systems Security - Sachin S. Shetty 2019-03-05
AN ESSENTIAL GUIDE TO USING BLOCKCHAIN TO PROVIDE FLEXIBILITY, COST-SAVINGS, AND SECURITY TO DATA MANAGEMENT, DATA ANALYSIS, AND INFORMATION SHARING Blockchain for Distributed Systems Security contains a description of the properties that underpin the formal foundations of Blockchain technologies and explores the practical issues for deployment in cloud and Internet of Things (IoT) platforms. The authors—noted experts in the field—present security and privacy issues that must be addressed for Blockchain technologies to be adopted for civilian and military domains. The book covers a range of topics including data provenance in cloud storage, secure IoT models, auditing architecture, and empirical validation of permissioned Blockchain platforms. The book's security and privacy analysis helps with an understanding of the basics of Blockchain and it explores the quantifying impact of the new attack surfaces introduced by Blockchain technologies and platforms. In addition, the

book contains relevant and current updates on the topic. This important resource: Provides an overview of Blockchain-based secure data management and storage for cloud and IoT Covers cutting-edge research findings on topics including invariant-based supply chain protection, information sharing framework, and trust worthy information federation Addresses security and privacy concerns in Blockchain in key areas, such as preventing digital currency miners from launching attacks against mining pools, empirical analysis of the attack surface of Blockchain, and more Written for researchers and experts in computer science and engineering, Blockchain for Distributed Systems Security contains the most recent information and academic research to provide an understanding of the application of Blockchain technology.

Introduction to Reliable and Secure Distributed Programming - Christian Cachin 2011-02-12

In modern computing a program is usually distributed among several processes. The fundamental challenge when developing reliable and secure distributed programs is to support the cooperation of processes required to execute a common task, even when some of these processes fail. Failures may range from crashes to adversarial attacks by malicious processes. Cachin, Guerraoui, and Rodrigues present an introductory description of fundamental distributed programming abstractions together with algorithms to implement them in distributed systems, where processes are subject to crashes and malicious attacks. The authors follow an incremental approach by first introducing basic abstractions in simple distributed environments, before moving to more sophisticated abstractions and more challenging environments. Each core chapter is devoted to one topic, covering reliable broadcast, shared memory, consensus, and extensions of consensus. For every topic, many exercises and their solutions enhance the understanding This book represents the second edition of "Introduction to Reliable Distributed Programming". Its scope has been extended to include security against malicious actions by non-cooperating processes. This important domain has become widely known under the name "Byzantine fault-tolerance".

Secure Programming with Static Analysis - Brian Chess 2007-06-29

The First Expert Guide to Static Analysis for Software Security! Creating secure code requires more than just good intentions. Programmers need to know that their code will be safe in an almost infinite number of scenarios and configurations. Static source code analysis gives users the ability to review their work with a fine-toothed comb and uncover the kinds of errors that lead directly to security vulnerabilities. Now, there’s a complete guide to static analysis: how it works, how to integrate it into the software development processes, and how to make the most of it during security code review. Static analysis experts Brian Chess and Jacob West look at the most common types of security defects that occur today. They illustrate main points using Java and C code examples taken from real-world security incidents, showing how coding errors are exploited, how they could have been prevented, and how static analysis can rapidly uncover similar mistakes. This book is for everyone concerned with building more secure software: developers, security engineers, analysts, and testers.

Introduction to Reliable Distributed Programming - Rachid Guerraoui 2006-05-01

In modern computing a program is usually distributed among several processes. The fundamental challenge when developing reliable distributed programs is to support the cooperation of processes required to execute a common task, even when some of these processes fail. Guerraoui and Rodrigues present an introductory description of fundamental reliable distributed programming abstractions as well as algorithms to implement these abstractions. The authors follow an incremental approach by first introducing basic abstractions in simple distributed environments, before moving to more sophisticated abstractions and more challenging environments. Each core chapter is devoted to one specific class of abstractions, covering reliable delivery, shared memory, consensus and various forms of agreement. This textbook comes with a companion set of running examples implemented in Java. These can be used by students to get a better understanding of how reliable distributed programming abstractions can be implemented and used in practice. Combined, the chapters deliver a full course on reliable distributed programming. The book can also be used as a complete reference on the basic elements required to build reliable distributed applications.

Building Secure and Reliable Network Applications - Kenneth P. Birman 1996

Distributed Computing - Hagit Attiya 2004-03-25

* Comprehensive introduction to the fundamental results in the mathematical foundations of distributed computing * Accompanied by supporting material, such as lecture notes and solutions for selected exercises * Each chapter ends with bibliographical notes and a set of exercises * Covers the fundamental models, issues and techniques, and features some of the more advanced topics

Designing Data-Intensive Applications - Martin Kleppmann 2017-03-16

Data is at the center of many challenges in system design today. Difficult issues need to be figured out, such as scalability, consistency, reliability, efficiency, and maintainability. In addition, we have an overwhelming variety of tools, including relational databases, NoSQL datastores, stream or batch processors, and message brokers. What are the right choices for your application? How do you make sense of all these buzzwords? In this practical and comprehensive guide, author Martin Kleppmann helps you navigate this diverse landscape by examining the pros and cons of various technologies for processing and storing data. Software keeps changing, but the fundamental principles remain the same. With this book, software engineers and architects will learn how to apply those ideas in practice, and how to make full use of data in modern applications. Peer under the hood of the systems you already use, and learn how to use and operate them more effectively Make informed decisions by identifying the strengths and weaknesses of different tools Navigate the trade-offs around consistency, scalability, fault tolerance, and complexity Understand the distributed systems research upon which modern databases are built Peek behind the scenes of major online services, and learn from their architectures