

## Distributed Computing

This book constitutes the refereed proceedings of the 5th International Conference on Parallel and Distributed Computing, Applications and Technologies; PDCAT 2004, held in Singapore in December 2004. The 173 papers presented were carefully reviewed and selected from 242 submissions. The papers focus on parallel and distributed computing from the perspectives of algorithms, networking and architecture, software systems and technologies, and applications. Besides classical topics from high performance computing, major recent developments are addressed, such as molecular computing, data mining, knowledge discovery, optical networks, secure computing and communications, wireless networks, mobile computing, component-based systems, Internet computing, and Web Technologies.

The 11th International Symposium on Distributed Computing and Artificial Intelligence 2014 (DCAI 2014) is a forum to present applications of innovative techniques for studying and solving complex problems. The exchange of ideas between scientists and technicians from both the academic and industrial sector is essential to facilitate the development of systems that can meet the ever-increasing demands of today's society. The present edition brings together past experience, current work and promising future trends associated with distributed computing, artificial intelligence and their application in order to provide efficient solutions to real problems. This year's technical program presents both high quality and diversity, with contributions in well-established and evolving areas of research (Algeria, Brazil, China, Croatia, Czech Republic, Denmark, France, Germany, Ireland, Italy, Japan, Malaysia, Mexico, Poland, Portugal, Republic of Korea, Spain, Taiwan, Tunisia, Ukraine, United Kingdom), representing a truly "wide area network" of research activity. DCAI'14 Special Sessions have been a very useful tool in order to complement the regular program with new or emerging topics of particular interest to the participating community. Special Sessions that emphasize on multi-disciplinary and transversal aspects, such as AI-driven methods for Multimodal Networks and Processes Modeling and Multi-Agents Macroeconomics have been especially encouraged and welcome. This symposium is organized by the Bioinformatics, Intelligent System and Educational Technology Research Group (<http://bisite.usal.es/>) of the University of Salamanca. The present edition was held in Salamanca, Spain, from 4th to 6th June 2014.

This book constitutes the refereed proceedings of the 20th International Symposium on Distributed Computing, DISC 2006. The book presents 35 revised full papers together with 1 invited paper and 13 announcements of ongoing works, all carefully selected for inclusion in the book. The entire scope of current issues in distributed computing is addressed, ranging from foundational and theoretical topics to algorithms and systems issues and to applications in various fields.

To understand the power of distributed systems, it is necessary to understand their inherent limitations: what problems cannot be solved in particular systems, or without sufficient resources (such as time or space). This book presents key techniques for proving such impossibility results and applies them to a variety of different problems in a variety of different system models. Insights gained from these results are highlighted, aspects of a problem that make it difficult are isolated, features of an architecture that make it inadequate for solving certain problems efficiently are identified, and different system models are compared. Table of Contents: Acknowledgments / Introduction / Indistinguishability / Shifting and Scaling / Scenario Arguments / Information Theory Arguments / Covering Arguments / Valency Arguments / Combinatorial Arguments / Reductions and Simulations / Bibliography / Authors' Biographies

Distributed Computing Principles, Algorithms, and Systems Cambridge University Press

Computers and computer networks are one of the most incredible inventions of the 20th century, having an ever-expanding role in our daily lives by enabling complex human activities in areas such as entertainment, education, and commerce. One of the most challenging problems in computer science for the 21st century is to improve the design of distributed systems where computing devices have to work together as a team to achieve common goals. In this book, I have tried to gently introduce the general reader to some of the most fundamental issues and classical results of computer science underlying the design of algorithms for distributed systems, so that the reader can get a feel of the nature of this exciting and fascinating field called distributed computing. The book will appeal to the educated layperson and requires no computer-related background. I strongly suspect that also most computer knowledgeable readers will be able to learn something new.

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

Distributed Computing by Mobile Entities is concerned with the study of the computational and complexity issues arising in systems of decentralized computational entities operating in a spatial universe Encompassing and modeling a large variety of application environments and systems, from robotic swarms to networks of mobile sensors, from software mobile agents in communication networks to crawlers and viruses on the web, the theoretical research in this area intersects distributed computing with the fields of computational geometry (especially for continuous spaces), control theory, graph theory and combinatorics (especially for discrete spaces). The research focus is on determining what tasks can be performed by the entities, under what conditions, and at what cost. In particular, the central question is to determine what minimal hypotheses allow a given problem to be solved. This book is based on the lectures and tutorial presented at the research meeting on "Moving and Computing" (mac)

held at La Maddalena Island in June 2017. Greatly expanded, revised and updated, each of the lectures forms an individual Chapter. Together, they provide a map of the current knowledge about the boundaries of distributed computing by mobile entities. This concise text is designed to present the recent advances in parallel and distributed architectures and algorithms within an integrated framework. Beginning with an introduction to the basic concepts, the book goes on discussing the basic methods of parallelism exploitation in computation through vector processing, super scalar and VLIW processing, array processing, associative processing, systolic algorithms, and dataflow computation. After introducing interconnection networks, it discusses parallel algorithms for sorting, Fourier transform, matrix algebra, and graph theory. The second part focuses on basics and selected theoretical issues of distributed processing. Architectures and algorithms have been dealt in an integrated way throughout the book. The last chapter focuses on the different paradigms and issues of high performance computing making the reading more interesting. This book is meant for the senior level undergraduate and postgraduate students of computer science and engineering, and information technology. The book is also useful for the postgraduate students of computer science and computer application.

This book constitutes the proceedings of the 30th International Symposium on Distributed Computing, DISC 2016, held in Paris, France, in September 2016. The 32 full papers, 10 brief announcements and 3 invited lectures presented in this volume were carefully reviewed and selected from 145 submissions. The focus of the conference is on following topics: theory, design, implementation, modeling, analysis, or application of distributed systems and networks.

This book constitutes the proceedings of the 11th International Conference on Internet and Distributed Computing Systems, IDCs 2018, held in Tokyo, Japan, in October 2018. The 21 full papers presented together with 5 poster and 2 short papers in this volume were carefully reviewed and selected from 40 submissions. This conference desired to look for inspiration in diverse areas (e.g., infrastructure and system design, software development, big data, control theory, artificial intelligence, IoT, self-adaptation, emerging models, paradigms, applications and technologies related to Internet-based distributed systems) to develop new ways to design and manage such complex and adaptive computation resources.

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

This book presents a collection of 38 position and research papers surveying the future landscape of research in distributed computing, written by the participants of the Workshop on Future Directions in Distributed Computing, held in Bertinoro, Italy in June 2002. The papers are grouped into four topical sections. The first deals with foundations of distributed computing. The second section surveys research issues in novel communication and network services. The third section is about data, file services, coherence, and replication in network computing. The last section deals with system and application issues. The book also includes two papers presenting insights into technological and social processes that are part of the development of the distributed computing technology. All in all, the book contains a plethora of research topics that are targets of future research or that are already being addressed by forward-looking research in distributed computing. The book was written to be a source of inspiration for researchers and a source of motivation for graduate students interested in entering the exciting research field of distributed computing.

A single dramatic software failure can cost a company millions of dollars - but can be avoided with simple changes to design and architecture. This new edition of the best-selling industry standard shows you how to create systems that run longer, with fewer failures, and recover better when bad things happen. New coverage includes DevOps, microservices, and cloud-native architecture. Stability antipatterns have grown to include systemic problems in large-scale systems. This is a must-have pragmatic guide to engineering for production systems. If you're a software developer, and you don't want to get alerts every night for the rest of your life, help is here. With a combination of case studies about huge losses - lost revenue, lost reputation, lost time, lost opportunity - and practical, down-to-earth advice that was all gained through painful experience, this book helps you avoid the pitfalls that cost companies millions of dollars in downtime and reputation. Eighty percent of project life-cycle cost is in production, yet few books address this topic. This updated edition deals with the production of today's systems - larger, more complex, and heavily virtualized - and includes information on chaos engineering, the discipline of applying randomness and deliberate stress to reveal systematic problems. Build systems that survive the real world, avoid downtime, implement zero-downtime upgrades and continuous delivery, and make cloud-native applications resilient. Examine ways to architect, design, and build software - particularly distributed

systems - that stands up to the typhoon winds of a flash mob, a Slashdotting, or a link on Reddit. Take a hard look at software that failed the test and find ways to make sure your software survives. To skip the pain and get the experience...get this book.

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  $\lambda$ -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.

The International Symposium on Distributed Computing and Artificial Intelligence 2013 (DCAI 2013) is a forum in which applications of innovative techniques for solving complex problems are presented. Artificial intelligence is changing our society. Its application in distributed environments, such as the internet, electronic commerce, environment monitoring, mobile communications, wireless devices, distributed computing, to mention only a few, is continuously increasing, becoming an element of high added value with social and economic potential, in industry, quality of life, and research. This conference is a stimulating and productive forum where the scientific community can work towards future cooperation in Distributed Computing and Artificial Intelligence areas. These technologies are changing constantly as a result of the large research and technical effort being undertaken in both universities and businesses. The exchange of ideas between scientists and technicians from both the academic and industry sector is essential to facilitate the development of systems that can meet the ever increasing demands of today's society. This edition of DCAI brings together past experience, current work, and promising future trends associated with distributed computing, artificial intelligence and their application in order to provide efficient solutions to real problems. This symposium is organized by the Bioinformatics, Intelligent System and Educational Technology Research Group (<http://bisite.usal.es/>) of the University of Salamanca. The present edition was held in Salamanca, Spain, from 22nd to 24th May 2013.

Many applications follow the distributed computing paradigm, in which parts of the application are executed on different network-interconnected computers. The extension of these applications in terms of number of users or size has led to an unprecedented increase in the scale of the infrastructure that supports them. Large-Scale Distributed Computing and Applications: Models and Trends offers a coherent and realistic image of today's research results in large scale distributed systems, explains state-of-the-art technological solutions for the main issues regarding large scale distributed systems, and presents the benefits of using large scale distributed systems and the development process of scientific and commercial distributed applications.

This timely text/reference describes the development and implementation of large-scale distributed processing systems using open source tools and technologies. Comprehensive in scope, the book presents state-of-the-art material on building high performance distributed computing systems, providing practical guidance and best practices as well as describing theoretical software frameworks. Features: describes the fundamentals of building scalable software systems for large-scale data processing in the new paradigm of high performance distributed computing; presents an overview of the Hadoop ecosystem, followed by step-by-step instruction on its installation, programming and execution; Reviews the basics of Spark, including resilient distributed datasets, and examines Hadoop streaming and working with Scalding; Provides detailed case studies on approaches to clustering, data classification and regression analysis; Explains the process of creating a working recommender system using Scalding and Spark.

\* 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

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.

This book contains a selection of refereed and revised papers of the Intelligent Distributed Computing Track originally presented at the third International Symposium on Intelligent Informatics (ISI-2014), September 24-27, 2014, Delhi, India. The papers selected for this Track cover several Distributed Computing and related topics including Peer-to-Peer Networks, Cloud Computing, Mobile Clouds, Wireless Sensor Networks, and their applications.

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](http://www.cambridge.org/9780521876346).

Distributed Computing is a textbook designed for students of computer science engineering, information technology, and computer applications. The book provides a clear understanding of the computing aspects of distributed systems.

Topics in Parallel and Distributed Computing provides resources and guidance for those learning PDC as well as those teaching students new to the discipline. The pervasiveness of computing devices containing multicore CPUs and GPUs, including home and office PCs, laptops, and mobile devices, is making even common users dependent on parallel processing. Certainly, it is no longer sufficient for even basic programmers to acquire only the traditional sequential programming skills. The preceding trends point to the need for imparting a broad-based skill set in PDC technology. However, the rapid changes in computing hardware platforms and devices, languages, supporting programming environments, and research advances, poses a challenge both for newcomers and seasoned computer scientists. This edited collection has been developed over the past several years in conjunction with the IEEE technical committee on parallel processing (TCPP), which held several workshops and discussions on learning parallel computing and integrating parallel concepts into courses throughout computer science curricula. Contributed and developed by the leading minds in parallel computing research and instruction Provides resources and guidance for those learning PDC as well as those teaching students new to the discipline Succinctly addresses a range of parallel and distributed computing topics Pedagogically designed to ensure understanding by experienced engineers and newcomers Developed over the past several years in conjunction with the IEEE technical committee on parallel processing (TCPP), which held several workshops and discussions on learning parallel computing and integrating parallel concepts

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.

This book constitutes the refereed proceedings of the First International Conference on Advances in Parallel, Distributed Computing Technologies and Applications, PDCTA 2011, held in Tirunelveli, India, in September 2011. The 64 revised full papers were carefully reviewed and selected from over 400 submissions. Providing an excellent international forum for sharing knowledge and results in theory, methodology and applications of parallel, distributed computing the papers address all current issues in this field with special focus on algorithms and applications, computer networks, cyber trust and security, wireless networks, as well as mobile computing and bioinformatics. Patterns and Skeletons for Parallel and Distributed Computing is a unique survey of research work in high-level parallel and distributed computing over the past ten years. Comprising contributions from the leading researchers in Europe and the US, it looks at interaction patterns and their role in parallel and distributed processing, and demonstrates for the first time the link between skeletons and design patterns. It focuses on computation and communication structures that are beyond simple message-passing or remote procedure calling, and also on pragmatic approaches that lead to practical design and programming methodologies with their associated compilers and tools. The book is divided into two parts which cover: skeletons-related material such as expressing and composing skeletons, formal transformation, cost modelling and languages, compilers and run-time systems for skeleton-based programming.- design patterns and other related concepts, applied to other areas such as real-time, embedded and distributed systems. It will be an essential reference for researchers undertaking new projects in this area, and will also provide useful background reading for advanced undergraduate and postgraduate courses on parallel or distributed system design.

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.

This book constitutes the refereed proceedings of the 17th International Conference on Distributed Computing, DISC 2003, held in Sorrento, Italy in October 2003. The 25 revised full papers presented were carefully reviewed and selected from 91 submissions. A broad variety of current issues in distributed computing is addressed, from foundational and theoretical issues to applications in various fields.

A lucid and up-to-date introduction to the fundamentals of distributed computing systems As distributed systems become increasingly available, the need for a fundamental discussion of the subject has grown. Designed for first-year graduate students and advanced undergraduates as well as practicing computer engineers seeking a solid grounding in the subject, this well-organized text covers the fundamental concepts in distributed computing systems such as time, state, simultaneity, order, knowledge, failure, and agreement in distributed systems. Departing from the focus on shared memory and synchronous systems commonly taken by other texts, this is the first useful reference based on an asynchronous model of distributed computing, the most widely used in academia and industry. The emphasis of the book is on developing general mechanisms that can be applied to a variety of problems. Its examples-clocks, locks, cameras, sensors, controllers, slicers, and synchronizers-have been carefully chosen so that they are fundamental and yet useful in practical contexts. The text's advantages include: Emphasizes general mechanisms that can be applied to a variety of problems Uses a simple induction-based technique to prove correctness of all algorithms Includes a variety of exercises at the end of each chapter Contains material that has been extensively class tested Gives instructor flexibility in choosing appropriate balance between practice and theory of distributed computing

The primary audience for this book are advanced undergraduate students and graduate students. Computer architecture, as it happened in other fields such as electronics, evolved from the small to the large, that is, it left the realm of low-level hardware constructs, and gained new dimensions, as distributed systems became the keyword for system implementation. As such, the system architect, today, assembles pieces of hardware that are at least as large as a computer or a network router or a LAN hub, and assigns pieces of software that are self-contained, such as client or server programs, Java applets or protocol modules, to those hardware components. The freedom she/he now has, is tremendously challenging. The problems alas, have increased too. What was before mastered and tested carefully before a fully-fledged mainframe or a closely-coupled computer cluster came out on the market, is today left to the responsibility of computer engineers and scientists invested in the role of system architects, who fulfil this role on behalf of software vendors and integrators, add-value system developers, R&D institutes, and final users. As system complexity, size and diversity grow, so increases the probability of inconsistency, unreliability, non responsiveness and insecurity, not to mention the management overhead. What System Architects Need to Know The insight such an architect must have includes but goes well beyond, the functional properties of distributed systems.

Distributed Computing provides an introduction to the core concepts and principles of distributed programming techniques. It takes a "how-to" approach where students learn by doing. Designed for students familiar with Java, the book covers programming paradigms, protocols, and application program interfaces (API's), including RMI, COBRA, IDL, WWW, and SOAP. Each chapter introduces a paradigm and/or protocol, and then presents the use of a DPI that illustrates the concept. The presentation uses narrative, code examples, and diagrams designed to explain the topics in a manner that is clear and concise. End-of-chapter exercises provide analytical as well as hands-on exercises to prompt the reader to practice the concepts and the use of API's covered throughout the text. Using this text, students will understand and be able to execute, basic distributed programming techniques used to create network services and network applications, including Internet applications.

A tutorial leading the aspiring Go developer to full mastery of Golang's distributed features. Key Features This book provides enough concurrency theory to give you a contextual understanding of Go concurrency It gives weight to synchronous and asynchronous data streams in Golang web applications It makes Goroutines and Channels completely familiar and natural to Go developers Book Description Distributed Computing with Go gives developers with a good idea how basic Go development works the tools to fulfill the true potential of Golang development in a world of concurrent web and cloud applications. Nikhil starts out by setting up a professional Go development environment. Then you'll learn the basic concepts and practices of Golang concurrent and parallel development. You'll find out in the new few chapters how to balance resources and data with REST and standard web approaches while keeping concurrency in mind. Most Go applications these days will run in a data center or on the cloud, which is a condition upon which the next chapter depends. There, you'll expand your skills considerably by writing a distributed document indexing system during the next two chapters. This system has to balance a large corpus of documents with considerable analytical demands. Another use case is the way in which a web application written in Go can be consciously redesigned to take distributed features into account. The chapter is rather interesting for Go developers who have to migrate existing Go applications to computationally and memory-intensive environments. The final chapter relates to the rather onerous task of testing parallel and distributed applications, something that is not usually taught in standard computer science curricula. What you will learn Gain proficiency with concurrency and parallelism in Go Learn how to test your application using Go's standard library Learn industry best practices with technologies such as REST, OpenAPI, Docker, and so on Design and build a distributed search engine Learn strategies on how to design a system for web scale Who this book is for This book is for developers who are familiar with the Golang syntax and have a good idea of how basic Go development works. It would be advantageous if you have been through a web application product cycle, although it's not necessary.

This book constitutes the refereed proceedings of the 19th International Conference on Distributed Computing, DISC 2005, held in Cracow, Poland, in September 2005. The 32 revised full papers selected from 162 submissions are presented together with 14 brief announcements of ongoing works chosen from 30 submissions; all of them were carefully selected for inclusion in the book.

The entire scope of current issues in distributed computing is addressed, ranging from foundational and theoretical topics to algorithms and systems issues and to applications in various fields.

This book brings together past insights, current research and promising future trends associated with distributed computing, artificial intelligence and their application in order to provide efficient solutions to real-world problems. The book is based on the International Conference on Distributed Computing and Artificial Intelligence 2020 (DCAI 2020), which provided a forum to present applications of innovative techniques for studying and solving complex problems in artificial intelligence and computing areas. It includes contributions on well-established and evolving areas of research, by authors from 26 countries, representing a truly "wide area network" of research activity

Distributed computing and Java go together naturally. As the first language designed from the bottom up with networking in mind, Java makes it very easy for computers to cooperate. Even the simplest applet running in a browser is a distributed application, if you think about it. The client running the browser downloads and executes code that is delivered by some other system. But even this simple applet wouldn't be possible without Java's guarantees of portability and security: the applet can run on any platform, and can't sabotage its host. Of course, when we think of distributed computing, we usually think of applications more complex than a client and server communicating with the same protocol. We usually think in terms of programs that make remote procedure

calls, access remote databases, and collaborate with others to produce a single result. Java Distributed Computing discusses how to design and write such applications. It covers Java's RMI (Remote Method Invocation) facility and CORBA, but it doesn't stop there; it tells you how to design your own protocols to build message passing systems and discusses how to use Java's security facilities, how to write multithreaded servers, and more. It pays special attention to distributed data systems, collaboration, and applications that have high bandwidth requirements. In the future, distributed computing can only become more important. Java Distributed Computing provides a broad introduction to the problems you'll face and the solutions you'll find as you write distributed computing applications. Topics covered in Java Distributed Computing: Introduction to Distributed Computing Networking Basics Distributed Objects (Overview of CORBA and RMI) Threads Security Message Passing Systems Distributed Data Systems (Databases) Bandwidth Limited Applications Collaborative Systems

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.

This book constitutes the refereed proceedings of the Second International Conference on Distributed Computing and Internet Technology, ICDCIT 2005, held in Bhubaneswar, India in December 2005. The 40 revised full papers and 19 revised short papers presented together with 2 invited plenary talks were carefully reviewed and selected from 426 submissions. Covering the main areas distributed computing, internet technology, system security, data mining, and software engineering the papers are subdivided in topical sections on network protocols, routing in mobile ad hoc network, communication and coverage in wireless networks, secured communication in distributed systems, query and transaction processing, theory of distributed systems, grid computing, internet search and query, e-commerce, browsing and analysis of Web elements, theory of secured systems, intrusion detection and ad hoc network security, secured systems techniques, software architecture, software optimization and reliability, formal methods, data clustering techniques, and multidimensional data mining.

To help researchers from different areas of science understand and unlock the potential of the Polish Grid Infrastructure and to define their requirements and expectations, the following 13 pilot communities have been organized and involved in the PLGrid Plus project: Acoustics, AstroGrid-PL, Bioinformatics, Ecology, Energy Sector, Health Sciences, HEPGrid, Life Science, Materials, Metallurgy, Nanotechnologies, Quantum Chemistry and Molecular Physics, and SynchroGrid. The book describes the experience and scientific results achieved by the project partners. Chapters 1 to 8 provide a general overview of research and development activities in the framework of the project with emphasis on services for different scientific areas and an update on the status of the PL-Grid infrastructure, describing new developments in security and middleware. Chapters 9 to 13 discuss new environments and services which may be applied by all scientific communities. Chapters 14 to 36 present how the PLGrid Plus environments, tools and services are used in advanced domain specific computer simulations; these chapters present computational models, new algorithms, and ways in which they are implemented. The book also provides a glossary of terms and concepts. This book may serve as a resource for researchers, developers and system administrators working on efficient exploitation of available e-infrastructures, promoting collaboration and exchange of ideas in the process of constructing a common European e-infrastructure.

[Copyright: a681d61f8b9a8f2a243a0b0004419807](https://doi.org/10.1007/978-3-642-00044-1)