

A Component Architecture For High Performance Scientific

Recent Advances in Parallel Virtual Machine and Message Passing Interface
Agile Project Management with Kanban
Speculative Execution in High Performance Computer Architectures
Recent Advances in Parallel Virtual Machine and Message Passing Interface
High-Level Data Fusion
High-Performance Computing
Formal Methods and Software Engineering
On-Chip Communication Architectures
Architecting Data-Intensive Applications
Understanding SCA (Service Component Architecture)
Knowledge Reuse and Agile Processes: Catalysts for Innovation
Component-Based Software Engineering
Designing Software-Intensive Systems: Methods and Principles
Architectures for Adaptive Software Systems
Proceedings of the IEEE International Symposium on High Performance Distributed Computing
High Performance Computing - HiPC 2006
Component-Based Software Engineering
High Performance Embedded Architectures and Compilers
Component-Based Software Engineering
Convergent Architecture
Reconstruction of Software Component Architectures and Behaviour Models Using Static and Dynamic Analysis
Testing and Quality Assurance for Component-based Software
Computer Systems: Architectures, Modeling, and Simulation
Software Architecture
Transactions on High-Performance Embedded Architectures and Compilers II
Objects, Components, Architectures, Services, and Applications for a Networked World
Architecture and Protocols for High-Speed Networks
Computational Science and Its Applications - ICCSA 2006
Proceedings
Enterprise JavaBeans Component Architecture
Recent Advances in Parallel Virtual Machine and Message Passing Interface
Service Oriented Architecture (SOA) For Dummies
A Parallel Algorithm Synthesis Procedure for High-Performance Computer Architectures
High Performance Embedded Architectures and Compilers
High-Performance Computing and Networking
Advances in Web-Age Information Management
A Process Algebraic Approach to Software Architecture Design
Concurrent Information Processing and Computing
Real-Time and Multi-Agent Systems
High Confidence Software Reuse in Large Systems

Recent Advances in Parallel Virtual Machine and Message Passing Interface

In the field of formal methods in computer science, concurrency theory is receiving a constantly increasing interest. This is especially true for process algebra. Although it had been originally conceived as a means for reasoning about the semantics of current programs, process algebraic formalisms like CCS, CSP, ACP, π -calculus, and their extensions (see, e.g., [154, 119, 112, 22, 155, 181, 30]) were soon used also for comprehending functional and nonfunctional aspects of the behavior of communicating concurrent systems. The scientific impact of process calculi and behavioral equivalences at the base of process algebra is witnessed not only by a very rich literature. It is in fact worth mentioning the standardization procedure that led to the development of the process algebraic language LOTOS [49], as well as the implementation of several modeling and analysis tools based on process algebra, like CWB [70] and CADP [93], some of which have been used in industrial case studies. Furthermore, process calculi and behavioral equivalences are by now

adopted in university-level courses to teach the foundations of concurrent programming as well as the model-driven design of concurrent, distributed, and mobile systems. Nevertheless, after 30 years since its introduction, process algebra is rarely adopted in the practice of software development. On the one hand, its technicalities often obfuscate the way in which systems are modeled. As an example, if a process term comprises numerous occurrences of the parallel composition operator, it is hard to understand the communication scheme among the various subterms. On the other hand, process algebra is perceived as being difficult to learn and use by practitioners, as it is not close enough to the way they think of software systems.

Agile Project Management with Kanban

Over the past decade, system-on-chip (SoC) designs have evolved to address the ever increasing complexity of applications, fueled by the era of digital convergence. Improvements in process technology have effectively shrunk board-level components so they can be integrated on a single chip. New on-chip communication architectures have been designed to support all inter-component communication in a SoC design. These communication architecture fabrics have a critical impact on the power consumption, performance, cost and design cycle time of modern SoC designs. As application complexity strains the communication backbone of SoC designs, academic and industrial R&D efforts and dollars are increasingly focused on communication architecture design. On-Chip Communication Architectures is a comprehensive reference on concepts, research and trends in on-chip communication architecture design. It will provide readers with a comprehensive survey, not available elsewhere, of all current standards for on-chip communication architectures. A definitive guide to on-chip communication architectures, explaining key concepts, surveying research efforts and predicting future trends Detailed analysis of all popular standards for on-chip communication architectures Comprehensive survey of all research on communication architectures, covering a wide range of topics relevant to this area, spanning the past several years, and up to date with the most current research efforts Future trends that will have a significant impact on research and design of communication architectures over the next several years

Speculative Execution in High Performance Computer Architectures

Multimedia data streams will form a major part of the new generation of applications in high-speed networks. Continuous media streams, however, require transmission with guaranteed performance. In addition, many multimedia applications will require peer-to-multipeer communication. Guaranteed performance can only be provided with resource reservation in the network, and efficient multipeer communication must be based on multicast support in the lower layers of the network. Architecture and Protocols for High-Speed Networks focuses on techniques for building the networks that will meet the needs of these multimedia applications. In particular two areas of current research interest in such communication systems

are covered in depth. These are the protocol related aspects, such as switched networks, ATM, MAC layer, network and transport layer; and the services and applications. Architecture and Protocols for High-Speed Networks contains contributions from leading world experts, giving the most up-to-date research available. It is an essential reference for all professionals, engineers and researchers working in the area of high-speed networks.

Recent Advances in Parallel Virtual Machine and Message Passing Interface

The only complete technical guide to building integrated business systems using the convergent architecture approach In his groundbreaking Business Engineering with Object Technology (0-471-04521-7), David Taylor introduced the concept of convergent architecture (CA), a framework for building the business design directly into the software systems that support it. Now, in this important follow-up to that 1995 classic, expert Richard Hubert provides systems developers and architects with their first complete blueprint for building integrated CA business systems using the hottest technologies, including Enterprise JavaBeans, XML, UML, Rational Rose, and others. Following a detailed introduction to the elements of CA, he walks readers through the entire CA design and implementation process, using examples in Java and EJB to illustrate key points. Companion Website provides hands-on tutorials, links to related tool sites, and updates to the CA methodology.

High-Level Data Fusion

High-Performance Computing

Providing all the latest on a topic of extreme commercial relevance, this book contains the refereed proceedings of the 10th International ACM SIGSOFT Symposium on Component-Based Software Engineering, held in Medford, MA, USA in July 2007. The 19 revised full papers presented were carefully reviewed and selected from 89 submissions. The papers feature new trends in global software services and distributed systems architectures to push the limits of established and tested component-based methods, tools and platforms.

Formal Methods and Software Engineering

On-Chip Communication Architectures

EJB allows developers to focus on the actual business architecture of their creations rather than worry about megalocs of

code. Because EJB systems are written in Java, they are platform independent and as they are object-oriented, they can be integrated into existing systems with little or no recompiling.

Architecting Data-Intensive Applications

caches. They introduce a reuse-distance drowsy cache mechanism that is simple as well as effective in reducing the static power in caches.

Understanding SCA (Service Component Architecture)

This book constitutes the refereed proceedings of the 7th International Symposium on Component-Based Software Engineering, CBSE 2004, held in Edinburgh, UK in May 2004 as an adjunct event to ICSE 2004. The 12 revised long papers and 13 revised short papers presented together with the abstracts of 2 invited talks were carefully reviewed and selected from 82 submissions. The papers are organized in topical sections on generation and adoption of component-based systems, tools and building frameworks, components for real-time embedded systems, extra-functional properties of components and component-based systems, and measurement and prediction models for component assemblies.

Knowledge Reuse and Agile Processes: Catalysts for Innovation

Component-Based Software Engineering

Despite five decades of research, parallel computing remains an exotic, frontier technology on the fringes of mainstream computing. Its much-heralded triumph over sequential computing has yet to materialize. This is in spite of the fact that the processing needs of many signal processing applications continue to eclipse the capabilities of sequential computing. The culprit is largely the software development environment. Fundamental shortcomings in the development environment of many parallel computer architectures thwart the adoption of parallel computing. Foremost, parallel computing has no unifying model to accurately predict the execution time of algorithms on parallel architectures. Cost and scarce programming resources prohibit deploying multiple algorithms and partitioning strategies in an attempt to find the fastest solution. As a consequence, algorithm design is largely an intuitive art form dominated by practitioners who specialize in a particular computer architecture. This, coupled with the fact that parallel computer architectures rarely last more than a couple of years, makes for a complex and challenging design environment. To navigate this environment, algorithm designers need a road map, a detailed procedure they can use to efficiently develop high performance, portable parallel

algorithms. The focus of this book is to draw such a road map. The Parallel Algorithm Synthesis Procedure can be used to design reusable building blocks of adaptable, scalable software modules from which high performance signal processing applications can be constructed. The hallmark of the procedure is a semi-systematic process for introducing parameters to control the partitioning and scheduling of computation and communication. This facilitates the tailoring of software modules to exploit different configurations of multiple processors, multiple floating-point units, and hierarchical memories. To showcase the efficacy of this procedure, the book presents three case studies requiring various degrees of optimization for parallel execution. This book can be used as a reference for algorithm designers or as a text for an advanced course on parallel programming.

Designing Software-Intensive Systems: Methods and Principles

Architectures for Adaptive Software Systems

This book constitutes the refereed proceedings of the 9th European PVM/MPI Users'Group Meeting held in Linz, Austria in September/October 2002. The 50 revised full papers presented together with abstracts of 11 invited contributions were carefully reviewed and selected. The papers are organized in topical sections on Corss Grid, Par Sim, application using MPI and PVM, parallel algorithms using message passing, programming tools for MPI and PVM, implementations of MPI and PVM, extensions of MPI and PVM, and performance analysis and optimization.

Proceedings of the IEEE International Symposium on High Performance Distributed Computing

This book constitutes the refereed proceedings of the 6th International Conference on Web-Age Information Management, WAIM 2005, held in Hangzhou, China, in October 2005. The 48 revised full papers, 50 revised short papers and 4 industrial papers presented together with 3 invited contributions were carefully reviewed and selected from 486 submissions. The papers are organized in topical sections on XML, performance and query evaluation, data mining, semantic Web and Web ontology, data management, information systems, Web services and workflow, data grid and database languages, agent and mobile data, database application and transaction management, and 3 sections with industrial, short, and demonstration papers.

High Performance Computing - HiPC 2006

Component-Based Software Engineering

Use Kanban to maximize efficiency, predictability, quality, and value With Kanban, every minute you spend on a software project can add value for customers. One book can help you achieve this goal: Agile Project Management with Kanban. Author Eric Brechner pioneered Kanban within the Xbox engineering team at Microsoft. Now he shows you exactly how to make it work for your team. Think of this book as “Kanban in a box”: open it, read the quickstart guide, and you’re up and running fast. As you gain experience, Brechner reveals powerful techniques for right-sizing teams, estimating, meeting deadlines, deploying components and services, adapting or evolving from Scrum or traditional Waterfall, and more. For every step of your journey, you’ll find pragmatic advice, useful checklists, and actionable lessons. This truly is “Kanban in a box”: all you need to deliver breakthrough value and quality. Use Kanban techniques to: Start delivering continuous value with your current team and project Master five quick steps for completing work backlogs Plan and staff new projects more effectively Minimize work in progress and quickly adjust to change Eliminate artificial meetings and prolonged stabilization Improve and enhance customer engagement Visualize workflow and fix revealed bottlenecks Drive quality upstream Integrate Kanban into large projects Optimize sustained engineering (contributed by James Waletzky) Expand Kanban beyond software development

High Performance Embedded Architectures and Compilers

This book constitutes the proceedings of the 7th European Conference on Software Architecture, ECSA 2013, held in Montpellier, France, in July 2013. The 25 full papers and 11 poster papers presented in this volume were carefully reviewed and selected from a total of 82 submissions. The contributions are organized in topical sections named: architectural and design patterns and models; ADLs and architectural MetaModels; architectural design decision-making; software architecture conformance and quality; and architectural repair and adaptation.

Component-Based Software Engineering

This book constitutes the refereed proceedings of the Fourth International Conference on High Performance Embedded Architectures and Compilers, HiPEAC 2009, held in Paphos, Cyprus, in January 2009. The 27 revised full papers presented together with 2 invited keynote paper were carefully reviewed and selected from 97 submissions. The papers are organized in topical sections on dynamic translation and optimisation, low level scheduling, parallelism and resource control, communication, mapping for CMPs, power, cache issues as well as parallel embedded applications.

Convergent Architecture

The book explores object and situation fusion processes with an appropriate handling of uncertainties, and applies cutting-edge artificial intelligence and emerging technologies like particle filtering, spatiotemporal clustering, net-centricity, agent formalism, and distributed fusion together with essential Level 1 techniques and Level 1/2 interactions.

Reconstruction of Software Component Architectures and Behaviour Models Using Static and Dynamic Analysis

Architect and design data-intensive applications and, in the process, learn how to collect, process, store, govern, and expose data for a variety of use cases

Key Features

- Integrate the data-intensive approach into your application architecture
- Create a robust application layout with effective messaging and data querying architecture
- Enable smooth data flow and make the data of your application intensive and fast

Book Description

Are you an architect or a developer who looks at your own applications gingerly while browsing through Facebook and applauding it silently for its data-intensive, yet fluent and efficient, behaviour? This book is your gateway to build smart data-intensive systems by incorporating the core data-intensive architectural principles, patterns, and techniques directly into your application architecture. This book starts by taking you through the primary design challenges involved with architecting data-intensive applications. You will learn how to implement data curation and data dissemination, depending on the volume of your data. You will then implement your application architecture one step at a time. You will get to grips with implementing the correct message delivery protocols and creating a data layer that doesn't fail when running high traffic. This book will show you how you can divide your application into layers, each of which adheres to the single responsibility principle. By the end of this book, you will learn to streamline your thoughts and make the right choice in terms of technologies and architectural principles based on the problem at hand.

What you will learn

- Understand how to envision a data-intensive system
- Identify and compare the non-functional requirements of a data collection component
- Understand patterns involving data processing, as well as technologies that help to speed up the development of data processing systems
- Understand how to implement Data Governance policies at design time using various Open Source Tools
- Recognize the anti-patterns to avoid while designing a data store for applications
- Understand the different data dissemination technologies available to query the data in an efficient manner
- Implement a simple data governance policy that can be extended using Apache Falcon

Who this book is for

This book is for developers and data architects who have to code, test, deploy, and/or maintain large-scale, high data volume applications. It is also useful for system architects who need to understand various non-functional aspects revolving around Data Intensive Systems.

Testing and Quality Assurance for Component-based Software

Until now, there were few textbooks that focused on the dynamic subject of speculative execution, a topic that is crucial to

the development of high performance computer architectures. Speculative Execution in High Performance Computer Architectures describes many recent advances in speculative execution techniques. It covers cutting-edge research

Computer Systems: Architectures, Modeling, and Simulation

The five-volume set LNCS 3980-3984 constitutes the refereed proceedings of the International Conference on Computational Science and Its Applications, ICCSA 2006. The volumes present a total of 664 papers organized according to the five major conference themes: computational methods, algorithms and applications high performance technical computing and networks advanced and emerging applications geometric modelling, graphics and visualization information systems and information technologies. This is Part II.

Software Architecture

This book constitutes the thoroughly refereed joint post-conference proceedings of the 6th International Symposium on High-Performance Computing, ISHPC 2005, held in Nara, Japan, in September 2005 and the First International Workshop on Advanced Low Power Systems 2006, ALPS2006, organized as satellite event of the 20th ACM International Conference on Supercomputing, ICS 2006, held in Cairns, Australia, in July 2006. The ISHPC 2005 symposium contributed 15 revised full papers and 13 revised short papers as well as 6 papers from the Second High Performance Fortran (HPF) International Workshop: Experiences and Progress, HiWEP 2005, and 8 papers from the Workshop on Applications for PetaFLOPS Computing, APC 2005 - all of which were carefully reviewed and selected from a total of 76 submissions. The volume is completed with 5 papers of the First International Workshop on Advanced Low Power Systems, ALPS2006, selected from 15 initial submissions that also passed a second round of reviewing and improvement. The papers are organized in topical sections on architecture of high performance computing, HPC applications or visualization, current HPF research, and low-power processing techniques in HPC.

Transactions on High-Performance Embedded Architectures and Compilers II

Much of a software architect's life is spent designing software systems to meet a set of quality requirements. General software quality attributes include scalability, security, performance or reliability. Quality attribute requirements are part of an application's non-functional requirements, which capture the many facets of how the functional requirements of an application are achieved. Understanding, modeling and continually evaluating quality attributes throughout a project lifecycle are all complex engineering tasks which continue to challenge the software engineering scientific community. While we search for improved approaches, methods, formalisms and tools that are usable in practice and can scale to large systems,

the complexity of the applications that the software industry is challenged to build is ever increasing. Thus, as a research community, there is little opportunity for us to rest on our laurels, as our innovations that address new aspects of system complexity must be deployed and validated. To this end the 5th International Conference on the Quality of Software Architectures (QoSA) 2009 focused on architectures for adaptive software systems. Modern software systems must often reconfigure their structure and behavior to respond to continuous changes in requirements and in their execution environment. In these settings, quality models are helpful at an architectural level to guide systematic model-driven software development strategies by evaluating the impact of competing architectural choices.

Objects, Components, Architectures, Services, and Applications for a Networked World

This book constitutes the refereed proceedings of the 12th European PVM/MPI Users' Group Meeting held in Sorrento, Italy in September 2005. The 61 revised full papers presented together with abstracts of 6 invited contributions were carefully reviewed and selected from numerous submissions. The papers are organized in topical sections on algorithms, extensions and improvements, cluster and grid, tools and environments, performance, applications and ParSim 2005.

Architecture and Protocols for High-Speed Networks

This book constitutes the refereed proceedings of the 8th International Conference on High-Performance Computing and Networking, HPCN Europe 2000, held in Amsterdam, The Netherlands, in May 2000. The 52 revised full papers presented together with 34 revised posters were carefully reviewed for inclusion in the book. The papers are organized in sections on problem solving environments, metacomputing, load balancing, numerical parallel algorithms, virtual enterprises and virtual laboratories, cooperation coordination, Web-based tools for tele-working, monitoring and performance, low-level algorithms, Java in HPCN, cluster computing, data analysis, and applications in a variety of fields.

Computational Science and Its Applications - ICCSA 2006

This book constitutes the thoroughly refereed post-proceedings of the international conference NetObjectDays 2002, held in Erfurt, Germany, in October 2002. The 26 revised full papers presented were carefully selected during two rounds of reviewing and revision. The papers are organized in topical sections on embedded and distributed systems; components and MDA; Java technology; Web services; aspect-oriented software design; agents and mobility; software product lines; synchronization; testing, refactoring, and CASE tools.

Proceedings

This highly relevant and up-to-the-minute book constitutes the refereed proceedings of the Third International Conference on High Performance Embedded Architectures and Compilers, HiPEAC 2008, held in Göteborg, Sweden, January 27-29, 2008. The 25 revised full papers presented together with 1 invited keynote paper were carefully reviewed and selected from 77 submissions. The papers are organized into topical sections on a number of key subjects in the field.

Enterprise JavaBeans Component Architecture

This is the refereed proceedings of the 9th International Symposium on Component-Based Software Engineering, CBSE 2006, held in Västerås, Sweden in June/July 2006. The 22 revised full papers and 9 revised short papers presented cover issues concerned with the development of software-intensive systems from reusable parts, the development of reusable parts, and system maintenance and improvement by means of component replacement and customization.

Recent Advances in Parallel Virtual Machine and Message Passing Interface

A detailed account of real-time systems, including program structures for real-time, phases development analysis, and formal specification and verification methods of reactive systems. The book brings together the 3 key fields of current and future data-processing: distributed systems and applications, parallel scientific computing, and real-time and manufacturing systems. It covers the basic concepts and theories, methods, techniques and tools currently used in the specification and implementation of applications and contains many examples plus complete case studies.

Service Oriented Architecture (SOA) For Dummies

Innovation, agility, and coordination are paramount in the support of value in the global knowledge economy. Therefore, the long-term success of a company is increasingly dependent on its underlying resilience and agility. Knowledge Reuse and Agile Processes: Catalysts for Innovation addresses flexibility of both business and information systems through component technology at the nexus of three seemingly unrelated disciplines: service-oriented architecture, knowledge management, and business process management. Providing practitioners and academicians with timely, compelling research on agile, adaptive processes and information systems, this Premier Reference Source will enhance the collection of every reference library.

A Parallel Algorithm Synthesis Procedure for High-Performance Computer Architectures

Feeling overwhelmed by the buzz about SOA—service oriented architecture? Take heart! Service Oriented Architecture For

Dummies, 2nd Edition makes it easy to understand, plan, and implement the latest SOA solutions for your business. Whether you're the IT person responsible for developing SOA or the executive who's trying to get a handle on the concept, Service Oriented Architecture For Dummies, 2nd Edition will help you understand what SOA is, why it's important, and how you can make the most of it. You'll find out about the business and financial aspects of SOA, how to decide if you need it, and what it can mean to your bottom line. Discover how to: Identify the main components of SOA and how they work to create business processes Create reusable, flexible systems and avoid common pitfalls Deconstruct business processes and applications to identify their components, then put them together in new ways Construct SOA business applications for maximum adaptability Confirm quality in a situation that's difficult to test, and assure the quality and consistency of your data Develop a governance strategy for SOA based on your company's philosophy and culture Work with XML and understand how it's used in SOA Maximize the benefits of unified communications Understand software ecosystems, rich interfaces, and the development lifecycle Packed with real-life case studies illustrating how SOA has been applied in a variety of industries, Service Oriented Architecture For Dummies, 2nd Edition demystifies one of today's hottest business tools.

High Performance Embedded Architectures and Compilers

Use SCA to Simplify the Development and Delivery of Service-Based Applications Service Component Architecture (SCA) is a new programming model that enables developers to build distributed applications more efficiently and effectively than previous technologies. In Understanding SCA (Service Component Architecture), two leading experts offer the first complete and independent guide to SCA. Drawing on extensive experience both developing the SCA standards and implementing large-scale SCA applications, Jim Marino and Michael Rowley provide an insider's perspective for developers and technical managers tasked with architecting and implementing enterprise systems. Rather than simply providing a technology overview, the authors draw on their practical experiences with SCA, explaining The full history behind SCA How SCA fits with other enterprise technologies such as JEE, .NET, Web Services, and BPEL All the major SCA concepts including composition, policy, wires, and bindings Best practices for designing SCA applications Using SCA with Web Services, Message-Oriented Middleware, BPEL, JPA, and Servlets Understanding SCA (Service Component Architecture) provides the background necessary to make informed decisions about when and how to best use SCA to build enterprise applications.

High-Performance Computing and Networking

From the basics to the most advanced quality of service (QoS) concepts, this all encompassing, first-of-its-kind book offers an in-depth understanding of the latest technical issues raised by the emergence of new types, classes and qualities of Internet services. The book provides end-to-end QoS guidance for real time multimedia communications over the Internet. It

offers you a multiplicity of hands-on examples and simulation script support, and shows you where and when it is preferable to use these techniques for QoS support in networks and Internet traffic with widely varying characteristics and demand profiles. This practical resource discusses key standards and protocols, including real-time transport, resource reservation, and integrated and differentiated service models, policy based management, and mobile/wireless QoS. The book features numerous examples, simulation results and graphs that illustrate important concepts, and pseudo codes are used to explain algorithms. Case studies, based on freely available Linux/FreeBSD systems, are presented to show you how to build networks supporting Quality of Service. Online support material including presentation foils, lab exercises and additional exercises are available to text adopters.

Advances in Web-Age Information Management

This book constitutes the refereed proceedings of the 10th International Conference on Software Reuse, ICSR 2008, held in Beijing, China, in May 2008. The 40 revised full papers presented together with 5 workshop summaries and 5 tutorials were carefully reviewed and selected from numerous submissions. The papers are organized in topical sections on architecture and reuse approaches, high confidence and reuse, component selection and reuse repository, product line, domain models and analysis, service oriented environment, components and services, reuse approaches and frameworks, as well as reuse approaches and methods.

A Process Algebraic Approach to Software Architecture Design

"This book addresses the complex issues associated with software engineering environment capabilities for designing real-time embedded software systems"--Provided by publisher.

Concurrent Information Processing and Computing

This book constitutes the refereed proceedings of the 4th International Conference on Formal Engineering methods, ICFEM 2002, held in Shanghai, China, in October 2002. The 43 revised full papers and 16 revised short papers presented together with 5 invited contributions were carefully reviewed and selected from a total of 108 submissions. The papers are organized in topical sections on component engineering and software architecture, method integration, specification techniques and languages, tools and environments, refinement, applications, validation and verification, UML, and semantics.

Real-Time and Multi-Agent Systems

This book constitutes the refereed proceedings of the 13th International Conference on High-Performance Computing, HiPC 2006, held in Bangalore, India in December 2006. The 52 revised full papers presented together with the abstracts of 7 invited talks were carefully reviewed and selected from 335 submissions. The papers are organized in topical sections on scheduling and load balancing, architectures, network and distributed algorithms, application software, network services, applications, ad-hoc networks, systems software, sensor networks and performance evaluation, as well as routing and data management algorithms.

High Confidence Software Reuse in Large Systems

This book constitutes the refereed proceedings of the 10th European PVM/MPI Users' Group Meeting held in Venice, Italy, in September/October 2003. The 64 revised full papers and 16 revised short papers presented together with abstracts of 8 invited contributions and 7 reviewed special track papers were carefully reviewed and selected from 115 submissions. The papers are organized in topical sections on evaluation and performance analysis; parallel algorithms using message passing; extensions, improvements, and implementations of PVM/MPI; parallel programming tools; applications in science and engineering; grid and heterogeneous computing; and numerical simulation of parallel engineering environments - ParSim 2003.

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