<<事务处理>>

图书基本信息

书名：<<事务处理>>

13位ISBN编号：9787115195869

10位ISBN编号：7115195862

出版时间：2009-5

出版时间：人民邮电出版社

作者：（美）格雷，（美）路透　著

页数：1070

前言

Why We Wrote this Book The purpose of this book is to give you an understanding of how large, distributed, hetero- geneous computer systems can be made to work reliablv. In contrast to the often complex methods of distributed computing, it presents a distributed system application development approach that call be used by mere mortals. WhY then doesn't the title use a term like dis- tributed systems, high reliability, interoperability,or client-server ?
Why use something as prosaic as transaction processing。
a term that for many people denotes old-fashioned, batch- oriented, mainframe-minded data processing ?
The point is-and that'S what makes the book SO long～that the design, implementation, and operation of large application systems, with thousands of terminals, employing hundreds of computers。
providing service with absolutely no downtime, cannot be done from a single perspective. An integrated（and integrating）perspective and methodology is needed to ap。
proach the distributed systems problem. Our gOal is to demonstrate that ffansactions provide this integrative conceptual framework. and that distributed transaction-oriented operating systems are the enabling technology. The client-server paradigm provides a good way of structuring the system and of developing applications. but it stiil needs transactions to con- trol the client. server interactions. In a nutshell：without transactions. distributed systems cannot be made to work for typical real. 1ife applications. This is not an outrageous claim；rather it is a lesson many people——system implemen- tors, system owners, and application developers-have learned the hard way. Of course, the concepts for building lurge systems have been evolving for a long time. In fact, some of the key ideas were developed way back when batch processing was in full swing, but they are far from being obsolete. Transaction processing concepts were conceived to master the corn- plexity in single-processor online applications. If anything, these concepts are even more critical now for the SUCCESSful implementation of massively distributed systems that work and fail in much more complex ways.

<<事务处理>>

内容概要

本书从系统的角度全面阐述事务处理的概念和技术，其中涉及终端上的表示管理、通信子系统、操作系统、数据库、程序设计语言的运行时系统以及应用开发环境等。
本书重点放在事务处理的基本概念上，主要阐述事务概念是如何用于解决分布式系统问题的，以及利用这些概念如何能够在有限的资金和风险范围内建立高性能、高可用性的应用系统。
全书重点讲述了事务处理基础、容错基础知识、面向事务的计算，并发控制、恢复、事务型文件系统、系统概览等7个主题，介绍了事务的ACID特性、并发的理论和实践、事务管理和恢复技术等方面的内容，最后还介绍了一个非常重要的资源管理器的实现。
本书主要面向计算机及相关专业的高年级本科生和研究生，适合作为事务处理导论、数据库系统、分布式系统、操作系统等课程的辅助教材，需要了解事务处理系统的开发人员也可将其作为基本参考书。

作者简介

Jim Gray （1944-2007）计算机科学大师，因在数据库和事务处理研究和实现方面的开创性贡献而获得1998年图灵奖。
美国科学院、工程院两院院士，ACM和IEEE两会会士：他25岁成为加州大学伯克利分校计算机科学学院第一位博士。
在IBM工作期间参与和主持了IMS、System R、SQUDS、DB2等项目的开发。
后任职于微软研究院．主要关注应用数据库技术来处理各学科的海量信息。
2007年1月独自驾船出海后失踪。

<<事务处理>>

书籍目录

章节摘录

插图：Key-sequenced placement stores the records sorted in key order. Key sequencing clusters related records together and allows sequential scanning of records in sorted order. Earlier sections of this chapter point out the benefits of this clustering and sequential ac- cess to data. When a new record amves. its key is computed and the record is placed near records witll related keys. Record insertion is a little expensive. but there are inge- nious algorithms that make it competitive with hashing. Given a key value, it is easy to find the record by using binary search on the file or by using some indexing structure. It is often desirable to associatively access a file via two different keys. For example. it is often convenient to access employees by either name or employee number . Suppose the employee records are stored in an associative file keyed by employee number（empno）. Then a second associative file, keyed by employee name（empname）, could store a record of the formfor each record in the employee file. By first looking in this second file under the empname key to find the empno, and then using this empno to associatively access the employee file, the system can fairly quickly find the desired employee record. Such index files are called secondary indices. It is often convenient to think of the direct address of a record as its key. If this is done. then secondary indices can be defined on direct files as well as on associative files . Most systems allow file designers to define many secondary indices on a base file. The file system automatically maintains the records in the secondary indices as records are in- serted into, updated in, and deleted from the primary file. Of course, the definition of the secondary index must be stored in the file descriptor. When a file is first opened. the descriptor is read bv the server. and all subsequent record operations on the file cause the relevant secondary indices to be used and maintained. 2. 5. 3 Distributed Files Parts of a file mav be distributed among servers in a computer network. This distribufion can take two forms：The files can be partitioned （fragments of the file are stored in different nodes）, or the files can be replicated（the whole contents of the files are stored at several nodes）. The definitions of partitioning and replication are fairly simple. A file is broken into fragments by declaring the key boundaries of each fragment：All records within that key range belong to that fragment. For example, if a file is keyed by sales region and customer number, then the file might be fragmented by region, with each region having a separate fragment. These fragments might be partitioned among the computers of the various regions, with each region storing the fragment for that region. In addition , all the fragments might be replicated at central headquarters. The descriptor of each fragment contaias a complete description of the entire file。

When a client opens the file, the file system looks at the descriptor and thereby knows about all the fragments . When the client issues a read-by-key, the request is dispatched to one of the servers managing that fragment . When the client issues a record insert, delete, or update Op. eration, the request is dispatched to all servers managing the fragment that holds the record. Associated secondary index reads and writes are handled similarlv .

<<事务处理>>

编辑推荐

事务处理广泛应用于数据库和操作系统等领域，对构建高性能、并发、分布式的可靠现代计算机系统至关重要。
《事务处理概念与技术(英文版)》是被誉为“事务处理圣经”的经典名著，由图灵奖得主Jim Gray和世界数据库权威AndreasReuter合著，是两位大师数十年学术研究和实践经验的结晶。
《事务处理概念与技术(英文版)》的组织和叙述方法独树一帜，作者将事务作为统一的概念框架．由此出发，笔锋所至，纵横开阖，引导读者从系统实现者的角度．全面深入地审视了计算机系统的方方面面．不仅阐述理论，而且针对各种实际问题，详细解释出现的原因，讲述大量已经在成功的商业和研究项目中经过验证、行之有效的事务处理实现技术，并提供了丰富的C语言代码。
书中处处闪烁着作者对计算机系统的渊博学识和真知灼见，无论你是程序员、架构师、数据库管理员，还是科研人员和高校师生，都将从《事务处理概念与技术(英文版)》中获益匪浅。

版权说明