

# 基于 P2P 的实化视图维护技术研究<sup>\*</sup>

左亚尧<sup>1,2</sup> 汤庸<sup>1</sup> 舒忠梅<sup>1</sup>

(中山大学计算机科学系 广州 510275)<sup>1</sup> (广东工业大学计算机学院 广州 510090)<sup>2</sup>

**摘要** 基于 P2P 的视图维护是一个重要的研究方向。当前的视图维护方法主要以传统的 C/S 模型为基础,当更新频繁时将会导致数据仓库超载而崩溃。针对这种现状,构建了基于 P2P 的主动视图维护模型,模型充分发掘 P2P 技术的潜在优势,克服了传统视图维护方法中数据仓库和数据源负载失衡及可伸缩瓶颈等问题。描述了 P2PAVM 的节点拓扑结构;重点阐述了 P2PAVM 的动态节点管理与路由等关键技术。

**关键词** Peer-to-Peer, 视图维护

## Research on P2P-based View Maintenance

ZUO Ya-Yao<sup>1,2</sup> TANG Yong<sup>1</sup> SHU Zhong-Mei<sup>1</sup>

(Department of Computer Science, Sun Yat-sen University, Guangzhou 510275)<sup>1</sup>

(Faculty of Computer, Guangdong University of Technology, Guangzhou 510090)<sup>2</sup>

**Abstract** View maintenance is an important research area in Peer-to-Peer technology. Currently, most view maintenance methods are based on the traditional C/S model, which will lead to the overload and crash at the data warehouse. To address the problems, the model of P2P-based active view maintenance (P2PAVM) is established, which can not only exploit the potential merits of P2P, but also overcome the problems of traditional view maintenance models (e. g., overload unbalance between data warehouse and data source, lacking of scalability). The topology of the peers in the P2PAVM is presented. The key issues of implementation are described, including dynamical management and routing of peers.

**Keywords** Peer-to-Peer, View maintenance

## 1 引言

数据仓库(Data Warehouse, DW)集成了多个分布式、自治或异构数据源上的信息,主要为查询和决策分析提供支持。随着数据仓库应用的不断扩展和深入,数据仓库从脱机维护到联机维护,从面向单用户到为多用户服务,是一个必然的过程。在这一发展过程中,数据仓库的实化视图维护是保证整个系统可用性和可靠性的一个关键问题。实化视图维护是指在数据仓库为用户提供服务的同时,当数据源的原始数据发生改变时,实时地将这种变化反映到数据仓库中,使相应的实化视图得到及时刷新。

实化视图是数据仓库中存储的主要信息实体,它将各分布式数据源的数据实体化。在一个典型的数据仓库系统中,数据源由不同的信息提供者拥有,其功能独立于其他数据源。一旦数据源产生更新,DW 必须通过视图维护算法来确保视图有效地进行相应的维护,以保证两者之间数据的一致性。当数据源产生更新之后,如何保持数据仓库与数据源的一致性在保证数据仓库中数据质量的一个关键问题。

## 2 相关工作

对视图进行维护时通常需要访问该视图本身所没有的数据,而访问分布式数据仓库系统中的基础数据源数据的代价是非常昂贵的。最近几年,许多学者从不同的角度提出了一些有效的视图维护方法,如延迟视图维护方法<sup>[1]</sup>、自维护方

法<sup>[2]</sup>,以及基于增量维护思想的 ECA 算法<sup>[3]</sup>、Strobe 算法<sup>[4]</sup>、在线纠错算法<sup>[5]</sup>、MDVM<sup>[6]</sup>和 PMDVM<sup>[7]</sup>等。这些算法大都以类似斯坦福大学的 WHIPS 模型<sup>[8]</sup>为研究平台,是一种典型的 C/S 结构。基于这种结构的视图维护模型对于 DW 计算能力范围内的顺序更新有较好的响应性能。但由于数据源具有自主性,不能对更新的数目及顺序作任何强制要求。因此这种模型以及基于这种模型之上的维护算法具有其固有的局限性,即并行处理能力十分弱,甚至无法并行处理,在产生大量更新的情形下会导致维护处理出现阻塞,影响数据仓库数据的一致性及数据质量。

与 DW 端的维护阻塞不同,在 C/S 模型中,大部分时间里很多客户机即 DS 端是空闲的,它们拥有大量没有使用的磁盘存储能力和计算能力。我们完全可以通过某种方式最大限度地使用这些计算机资源。随着对等连接和资源有效利用技术的逐步深入研究,对等计算(Peer-to-Peer,简称 P2P)被认为是未来重构分布式体系结构的关键技术<sup>[9]</sup>。

通过对视图维护和 P2P 两方面的初步研究,我们认为,在数据仓库的视图维护思想中引入 P2P 的技术模型和原理有助于另辟蹊径,从根本上突破目前视图维护效率的瓶颈。文<sup>[10]</sup>将 P2P 思想合理地引入到视图维护的体系结构中,构建了基于 P2P 的分布式并行视图维护模型 P2PAVM。本文将在在此基础上,进一步对 P2PAVM 的维护机制、动态节点管理等关键技术进行深入研究。这些工作对于保证数据仓库中数据质量以及某些应用领域对数据的特殊时态性要求等,都

<sup>\*</sup>本文得到国家自然科学基金项目(60373081)、广东省自然科学基金重点项目(04105503,05200302)资助。

具有十分重要的理论意义和应用价值。

### 3 问题描述

考虑视图:  $MV = \prod_{X \in C(Y)} (R_1 \times R_2 \times \dots \times R_n)$ , 令  $F(R_i)$  为关系模式  $R_i$  中所有属性变量的集合,  $C(Y)$  是表示限制条件的布尔表达式,  $X$  和  $Y$  是 SPJ 表达式中关系模式  $R_1, R_2, \dots, R_n$  的属性变量的集合, 有  $X, Y \subseteq \bigcup_{i=1}^n F(R_i)$ 。  $R_1, R_2, \dots, R_n$  是  $n$  个分别位于数据源  $DS_1, DS_2, \dots, DS_n$  上的关系, 数据源  $DS_1, DS_2, \dots, DS_n$  和实化视图  $MV$  都分布在网络的不同主机上。

当其中一个关系  $R_i$  产生更新  $\Delta R_i$  后, 按照增量式视图维护方法,  $MV \leftarrow MV \oplus \Delta MV$ , 其中  $\oplus = \{U, -\}$ 。可通过使用关系代数表达式对更新  $\Delta R_i$  的操作类型进行描述: 当更新  $\Delta R_i$  为插入操作时表示为:  $R_i \leftarrow R_i \cup E$ ,  $E$  是关系代数表达式; 当更新  $\Delta R_i$  为删除操作时表示为:  $R_i \leftarrow R_i - E$ 。由于  $MV$  已存在, 故只需计算:

$$\Delta MV = \prod_{X \in C(Y)} (R_1 \times R_2 \times \dots \times \Delta R_i \times R_n) \quad (1)$$

目前, 大多数实化视图维护方法所使用的模型如图 1 所示。当  $DS_1$  上的  $R_1$  产生更新  $\Delta R_1$  后, 它将主动通知 DW (假设  $n=3$ , 如图中的步骤 1), DW 查找受该更新影响的视图并生成相应的维护任务  $\Delta MV$ , 然后依次联系其它各相关 DS 对该维护任务进行计算 (如图中的步骤 2~5), 从而得到最后的维护结果。

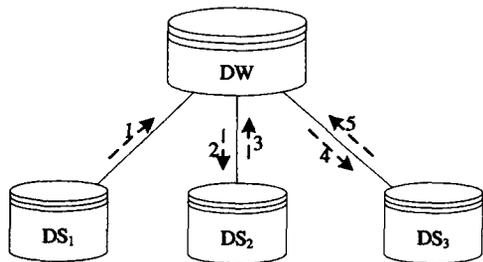


图 1 传统的数据仓库模型

在上述维护方式中, 主要存在两个问题: 其一, 各 DS 只是提供了信息的存储及主动将更新报告给 DW 的能力, 所有与查询、更新维护相关的计算都由 DW 端全权负责, 当更新频繁时将会导致 DW 超载而崩溃, 如图 2 所示; 其二, 维护任务  $\Delta MV$  的计算过程是按顺序方式进行的 (如图 1 中的步骤 2~4), 当视图所涉及的 DS 及其基本关系数目较大时, 这种顺序计算的开销是非常巨大而且是难以接受的。

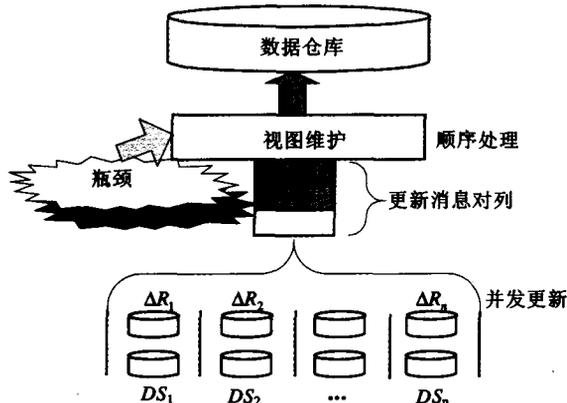


图 2 实化视图维护瓶颈

### 4 基于 P2P 的实化视图维护

#### 4.1 实化视图维护拓扑结构

针对上述两个问题, 我们提出了基于 P2P 的并行解决方案。由于视图维护模型主要拥有两类节点: DS 节点和 DW 节点, 从节点分布的拓扑结构角度分析, 不难发现导致 WHIPS 模型弱并行能力的本质原因在于负载失衡。在 C/S 结构的模型中, DW 节点与 DS 节点之间的负载关系为  $1:n$ , 即当 DW 节点疲于响应  $n$  个 DS 节点的维护信息时, 大量的 DS 节点可能出于空载状态。

为此, 为了克服 DW 和 DS 负载失衡问题, 最大程度地发挥 WHIPS 系统中所有节点的计算能力, 充分挖掘 DW 快速查询响应、决策支持等高端应用的能力, 我们构建了基于 P2P 环境下的实化视图维护模型 P2PAVM [10]。P2PAVM 模型的主要出发点就是从均衡各个节点负载和均衡 DW 维护模块负载的两个层面入手, 为 WHIPS 模型引入对等计算的思想, P2PAVM 系统将视图维护模块分别部署在 DW 节点和 DS 节点上, 让每个 DS 节点除了提供数据存储功能之外, 还能进行查询和维护工作。发起更新的 DS 节点将根据 DW 节点提供的相关信息产生相应的维护查询, 并自主地联系其他 DS 节点, 协同完成查询和更新维护工作, 并将维护后的结果主动发送给 DW 节点。而 DW 节点主要负责收集和记录视图定义、DS 节点信息等元数据, 从而使得 DW 节点在很大程度上从数据集成功能中解脱出来, 进而更好地响应数据分析与数据挖掘等计算需求。同时, 为了保持节点的自治性, 系统中的 DS 节点只需对其衍生关系的更新主动发起维护, 有效地实施了计算的分布性与负载的均衡性, 扩充并行计算能力的目的, 其拓扑结构如图 3 所示。

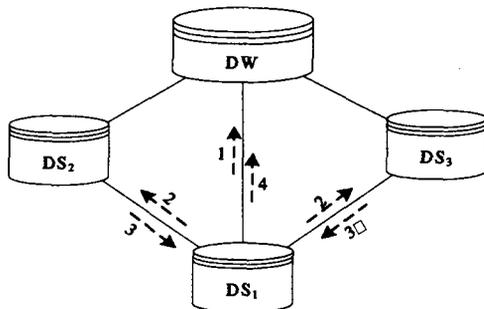


图 3 基于 P2P 的实化视图维护拓扑

对于维护任务按顺序重新计算的问题, 我们在增量式视图维护的基础上, 提出了分解维护任务的方法, 以并行的方式提高维护实化视图的效率 [11]。其核心思想在计算维护查询任务  $\Delta MV$  时, 首先将维护查询  $\Delta MV$  分解为左右子查询, 即  $\Delta MV = \Delta MV_{left} \times \Delta MV_{right}$ 。  $\Delta MV_{left}$  和  $\Delta MV_{right}$  的计算可以独立进行, 因而可以并行地执行。这只需要在  $\Delta MV_{left}$  和  $\Delta MV_{right}$  都计算结束后, 将两个结果进行合并。当  $DS_1$  上的  $R_1$  产生更新  $\Delta R_1$  后其维护过程如图 3 所示。

#### 4.2 节点管理与路由

在 P2P 网络系统中, 节点的路由和资源发现是非常关键而且棘手的问题, 同时也是 P2P 系统的共性问题。不仅网络计算通过它完成计算请求的路由和转发, 将计算请求传递给满足此请求的分布式计算资源, 而且点对点的文件共享、检索系统也依赖它提供资源信息发现以及资源之间的动态组合。

从机制上而言, 节点路由有广播方法和分布式 Hash 表

DHT方法,前者如 Gnutella, Freenet, KazaA, Morpheus 等系统,但该方法大部分情况下效率较低,定位稀疏资源困难;后者如 CAN, Tapestry, Chord, Pastry 等系统, DHT 方法的核心是在资源的索引编码与 P2P 网络节点的标识码之间建立相对一致的映射,目前基于 DHT 的方法应用较广泛<sup>[12]</sup>。

在数据仓库系统中, DS 节点资源相对 DW 节点而言是已知的,视图的定义是基于已知的 DS 资源及其衍生的关系,对于与 DW 视图定义无关的节点,数据仓库系统视而不见。典型情况下,这些节点的信息是静态存储在 DW 的元数据存储库中。而且在 P2PAVM 系统中,不仅利用了 P2P 网络完成资源共享问题,同时更侧重于它的并行计算能力方面。因此,尽管 P2PAVM 也会涉及到节点的路由和资源定位问题,但由于视图维护的特殊的工作机制,它对于节点的信息是集中式管理的。结合对系统可扩展性的考虑,我们采用了集中式索引与 DHT 方式相结合的方法来管理所有节点资源。

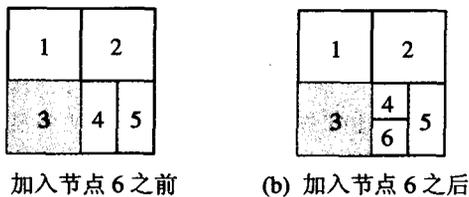


图 4 二维的 CAN 拓扑

(1) 动态节点管理 CAN<sup>[13]</sup>网络是在物理网络上定义的一个点对点的逻辑网络层。每个节点在 CAN 中都有一个虚拟的 P2P 地址,整个网络的逻辑拓扑是一个  $n$  维的笛卡尔空间。如图 4(a)所示,每个节点对应  $n$  维空间中的一个子空间,所有节点构成了  $n$  维空间的一个划分。每个节点的 P2P 地址是随网络中节点的加入而不断增长的,网络中只有一个节点的情况下,该节点地址是 0;每当有新节点  $N_1$  加入时,需要对某个已有节点  $N_0$  的对应的子空间进行二分,分解得到的两个空间一个分配给  $N_1$ ,另一个留给原来的节点  $N_0$ ,且  $N_1$  的地址是原  $N_0$  的地址末尾添 1,  $N_0$  的地址末尾添 0,从而保证节点地址是随网络状况增长的。如图 4(a)所示,节点

4 地址是 100,加入新节点 6 后空间节点的分布情况如图 4(b)所示,节点 4 地址变为 1000,节点 6 地址为 1001。

此外,为了标识某个已有的节点在某个时刻是否处于活动状态,即是否在线上,我们给每个子空间赋予颜色属性:白色(表示所对应的节点是活动的)和灰色(表示所对应的节点是非活动的)。例如图 4 中,节点 3 所对应的子空间为灰色,意味着当时节点 3 不在线上,而其它节点都在线上。

CAN 是 DHT 的一种,能够在  $O(\lg n)$  跳之内定位到索引对应的节点。CAN 网络中每个节点都维护着和邻居节点的连接信息,节点的邻居就是  $n$  维空间中的邻接空间对应的节点。路由到具体 P2P 地址的节点的方法,就是从出发点开始,沿着  $n$  维空间中的最短路径到达目的点。

(2) 集中式索引及搜索技术 在 P2PAVM 模型中,对更新  $\Delta R_i$  的相关视图集的一致性维护可以通过分布节点来并行处理,因此在 P2PAVM 中,当一个更新  $\Delta R_i$  产生后,如何有效搜索受该更新影响的相关视图集  $RMV(R_i)$  是一个亟待解决的问题,也是维护处理展开的前提。视图  $MV = \prod_X \sigma_{C(X)}$  ( $R_1 \times R_2 \times \dots \times R_n$ ) 可以看作一个有向图,本文采用十字链表结构来管理视图与关系间的关系以及搜索  $RMV(R_i)$ 。

如图 5(a)给出了一个十字链表示意图,图中有顶点结点和弧结点两类结点,顶点结点也就是视图定义首结点,结点的结构如图 5(b)。弧结点表示视图与关系的关联关系,它有四个域:弧头域  $R_{ID}$  和弧尾域  $V_{ID}$  分别是关系标识符和视图标识符,表明了关系  $R_i$  与视图  $V_j$  之间是否相关,链域 *Down* 指向与弧头域相同的下一条弧,而链域 *Right* 指向与弧尾域相同的下一条弧。弧头相同的弧在同一链表上,它们的集合就是弧头  $R_i$  的相关视图集  $RMV(R_i)$ ,具有相同弧尾  $V_j$  的弧也在同一年链表上,它们是构成视图  $V_j$  的所有源关系的集合。所有弧结点的头结点即为顶点结点,顶点结点管理视图定义中所涉及到的全部关系,它包含视图标识  $V_{ID}$  域、视图维护类型 VT 域以及 *Right* 域, *Right* 域指向以  $V_{ID}$  域为弧尾的第一个弧结点。

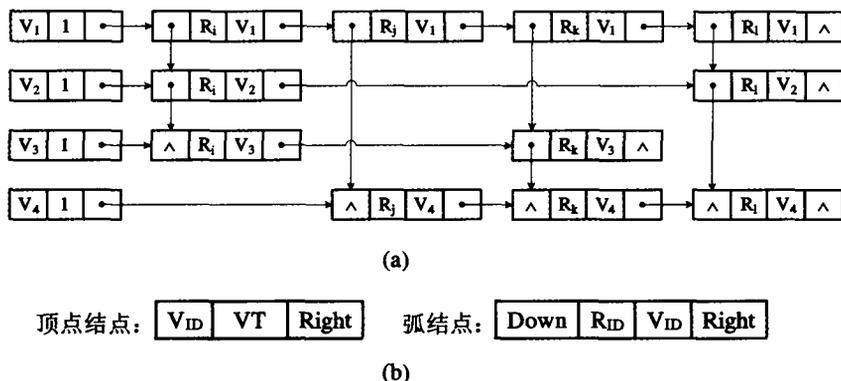


图 5 视图和源关系关联的十字链表

整个链表由水平链和垂直链两类链组成,水平链指出了视图维护的类型 VT 以及定义所包含的全部关系,垂直链表明一个关系的所有相关视图。水平链表初始化时,系统检查有向图的弧结点,如果视图  $V_j$  定义中涉及关系  $j$ ,那么存在弧结点,  $\boxed{Down \ R_i \ V_j \ Right}$ ,且令水平链表的上一弧结点

的 *Right* 域指向本结点。如果  $V_i$  与关系  $j$  无关,则水平链表的上一弧结点的 *Right* 域直接指向下一相关的弧结点  $\boxed{Down \ R_i \ V_j \ Right}$  (假定  $V_i$  定义中涉及关系  $k$ )。同理,在垂直链表上,如果  $R_i$  参与了视图  $V_j$  的关系运算,那么存在

(下转封四)

(上接第 94 页)

弧结点  $\boxed{Down \mid R_i \mid V_j \mid Right}$ , 且令垂直链表的上一弧结点的  $Down$  域指向本结点。如果  $R_i$  不是定义视图  $V_j$  的关系代数中的源关系, 则垂直链表的上一弧结点的  $Down$  域直接指向下一相关的弧结点  $\boxed{Down \mid R_i \mid V_k \mid Right}$  (假定  $R_i$  参与了视图  $V_i$  的关系运算)。

在 P2PAVM 模型中,  $DS_i$  节点上源关系  $R_i$  产生的更新消息传播到 DW 中, 集成器更新消息进行解析后, 以  $R_i$  为主键对视图定义的十字链表进行纵向搜索, 找出  $RMV(R_i)$ 。进而以  $MV_j (MV_j \in RMV(R_i))$  为主键在元数据存储器的视图树中进行匹配搜索, 获得定义视图  $MV_j$  的元数据。

## 5 性能分析

在 P2PAVM 视图维护模型中, 随着 DS 节点的不增加, 整个系统的处理能力将随之增强而不是减弱; DW 的处理能力也不会因为 DS 节点的规模而超载; 而且, DW 节点和 DS 节点能最大程度地并行处理。其主要原因在于 P2PAVM 的对等计算思想为我们带来了负载的平衡和不同级别的并行:

(1) 节点级并行。当有多个 DS 节点同时产生更新后, 各个更新节点可同时与 DW 节点通信。获得受各自更新影响的相关视图及其关系所在的维护节点信息后, 各更新节点分别主动与其维护节点进行联系, 在更新节点得到全局维护结果后才转发给 DW 节点对相关视图进行相应维护。这些更新节点对更新的维护工作是自主进行的, 完全可以并行处理。

(2) 维护级并行。针对单个 DS 节点上的单个更新而言, 由于更新节点的视图管理器在获得 DW 节点发来的相关信息后, 会根据视图定义产生一个全局维护查询。然后根据视图参与者之间的关联对该查询进行分解, 得到一系列的维护子查询。这些维护子查询随后转发给相应的维护节点, 这些维护节点在处理各自的维护子查询时也可以并行处理。

实验采用 JXTA 作为开发平台、SQL SERVER 作为底层数据库管理系统, 搭建了 P2PAVM 架构及维护方法的使用平台, 其性能分析见文[14]。

**结束语** 本文简要地介绍了现有的视图维护方法和模型方面的现状及其局限性, 在构建 P2PAVM 主动视图维护模型的基础上, 介绍了其节点拓扑结构, 剖析了 P2PAVM 实现的关键技术; 结果表明 P2PAVM 在并行处理方面具有优良

特性。基于 P2P 的视图维护是一个新的研究课题, 我们将在今后深入研究节点对维护查询的分解、系统的容错机制及安全等问题。

## 参考文献

- 1 Kawaguchi, Lieuwen D, Mumick I, et al. Concurrency control theory for deferred materialized views. In: the International Conference on Database Theory, Delphi, Greece, 1997. 306~320
- 2 Gupta, Jagadish H, Mumick I. Data integration using self-maintainable views. In: Proc. of Fifth Intl. Conf. on Extending Database Technology, Avignon, France, 1996. 140~144
- 3 Zhung, Garcia-Molina Y, Hammer H, et al. View maintenance in a warehousing environment. In: Proc. of the ACM SIGMOD Intl. Conf. on Management of Data, 1995. 316~327
- 4 Zhuge, Garcia-Molina Y, Wiener H. The Strobe algorithms for multi-source warehouse consistency. In: Proc. of the Fourth PDIS Conference, Miami Beach, Florida, Dec. 1996
- 5 Agrawal D, Abbadi A, Singh A, et al. Efficient view maintenance at data warehouses. In: ACM SIGMOD Conference, Tucson, Arizona, USA. 1997. 417~427
- 6 左亚尧, 舒忠梅, 潘久辉. 一种高效的视图维护算法. 计算机研究与发展, 2003, 40(4): 627~633
- 7 舒忠梅, 李师贤, 左亚尧. 并行多维视图维护方法. 计算机研究与发展, 2004, 41(增刊): 1~8
- 8 Wiener J L, Gupta H, et al. A System Prototype for Warehouse View Maintenance. In: Workshop on Materialized Views: Techniques and Applications, 1996. 26~33
- 9 Ng W S, Ooi B C, Tan K L. BestPeer: A self-configurable peer-to-peer system. In: Chrysanthis P K, ed. Proc. of the 18th ICDE. San Jose: IEEE Computer Society Press, 2002
- 10 舒忠梅, 李师贤, 等. 基于 Peer-to-Peer 的主动视图维护模型研究. 计算机科学, 2005, 32(7 suppl): 344~348
- 11 舒忠梅, 李师贤, 等. 基于维护查询分解的实化视图并行维护方法. 中国计算机大会, 2005
- 12 程学旗, 吕建明, 周昭涛. 基于对等网络的全文信息检索. 计算机研究与发展, 2004, 41(12): 2148~2155
- 13 Ratnasamy S, Francis P, Handley M, et al. A Scalable Content-Addressable Network. ACM SIGCOMM'01, San Diego, CA, 2001
- 14 舒忠梅. 基于 P2P 的视图维护架构及方法研究: [中山大学博士论文]. 2005

# 计算机科学

(1974年1月创刊)

第33卷第7期(月刊)

2006年7月25日出版

国际标准连续出版物号 ISSN 1002-137X  
国内统一连续物出版号 CN50-1075/TP

定价: 30.00元 国外定价: 5美元

邮发代号: 78-68

发行范围: 国内外公开

主管单位: 国家科学技术部

主办单位: 国家科技部西南信息中心

编辑出版: 《计算机科学》杂志社

重庆市渝中区胜利路132号 邮政编码: 400013

电话: (023) 63500828 E-mail: jsjcx@swic.ac.cn

网址: www.jsjcx.com

社长: 牟炳林

总编: 彭丹

主编: 朱宗元

主编助理: 徐书令

印刷者: 重庆科情印务有限公司

总发行处: 重庆市邮政局

订购处: 全国各地邮政局

国外总发行: 中国国际图书贸易总公司(北京399信箱)

国外代号: 6210-MO