

时空数据库索引机制研究^{*}

黄曙荣^{1,2} 秦小麟¹

(南京航空航天大学信息科学与技术学院 南京210016)¹ (盐城工学院计算机工程系 江苏盐城224003)²

Research of Index Mechanisms in Spatio-temporal Databases

HUANG Shu-Rong^{1,2} QIN Xiao-Lin¹

(College of Information Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing Jiangsu 210016)¹

(Department of Computer Engineering, Yancheng Institute of Technology, Yancheng Jiangsu 224003)²

Abstract Spatio-temporal database manages the large amount of spatial objects that change over time. It is necessary to query the spatio-temporal objects of the past and the current and to anticipate the future of spatio-temporal objects. It is important to design an efficient index mechanism for accessing the spatio-temporal data efficiently. The paper analyzes the features of the spatio-temporal objects, studies the methods of spatio-temporal index mechanisms, classifies the index mechanisms, and discusses the key technologies of spatio-temporal indexes. And it also presents the index methods of STADBS that we are studying.

Keywords Spatio-temporal databases, Spatio-temporal objects, Spatio-temporal indexes, STAM

1 引言

时空数据库(Spatio-temporal Database)是指存储随着时间发生变化的空间对象的数据库系统,这里的变化包括对象的位置和/或范围的变化。时空数据库的应用涉及到全球变化(如气候、陆地的变化)、交通运输(交通监控、智能传输系统)、社会(人口统计、健康)及多媒体系统等多个方面。

因为涉及到时间因素,时空数据库需要管理大量的带有长期时间信息的空间数据,为了对这些数据进行有效存取,对时空数据库索引机制的研究显得尤为重要。但迄今为止,对时空数据库的研究仍处于起步阶段,且主要集中于对时空模型和查询语言的研究,在时空索引方面的研究还很有限。一方面,对空间索引的研究成果可以支持对点、任意形状的对象及光栅数据的查询操作,但没有考虑到时空应用中的时间特性;另一方面,已提出的时间存取方法只对有效时间(valid-time)和/或事务时间(transaction-time)进行索引以支持时间数据库的索引,没有考虑到时空对象的空间属性。

本文分析了时空对象的特点,研究了时空索引机制的方法并进行了分类,讨论了时空索引中的关键技术,并对我们正在研制的时空分析数据库管理系统 STADBS 中的索引方法作了介绍。

2 时空对象(spatio-temporal object)

2.1 时空对象的概念

空间位置和/或范围随着时间的变化而发生变化的空间对象称为时空对象(也称为移动对象),如正在飞行的飞机、高速公路上行驶的小汽车、随着时间的变化其边界范围在扩大或缩小的森林等等。有大量的应用需要存储和管理时空对象,查找对象在过去、现在的位置和范围,推测其在未来某一时间戳(time-stamp)或时间间隔(time-interval)的行为。

时空数据库存储的是随时间变化的空间对象。在空间数据库中定义了的三种基本空间对象:点(points)、线(lines)和区域(regions)。点表示空间范围为零或只关心其空间位置,而不是它的大小的空间对象,例如在一张大比例的地图中的一个城市;线即空间曲线,可描述在空间中移动的设施或者空间连线,例如道路、河流、电话线、电线等;具有一定空间范围的对象称为区域,例如一个国家行政区、一片湖泊等。因此,目前对时空数据类型的研究主要集中于移动点(moving points)和移动区域(moving regions)这二种基本时空对象数据,由于线(曲线)本身就是运动的抽象或者投影,可转换为对移动点轨迹(trajjectory)的研究。

二维参考空间(x,y)中的点对象随时间变化情况可用三维参考空间的曲线表示,区域对象的变化不仅包括空间位置的改变,还包括形状的变化,如扩张、收缩等。图1(a)、(b)分别给出了二维参考空间中移动点和移动区域的例子(第三维是时间维)。本文讨论的是二维参考空间中的空间对象,同样的讨论也可扩展到三维或更高维的空间。

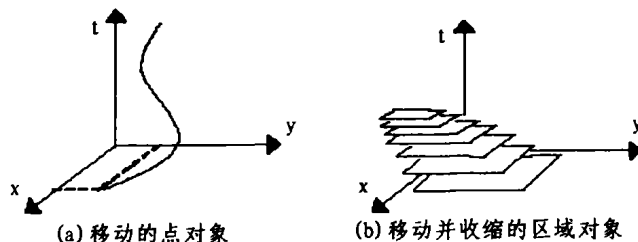


图1 二维参考空间中的移动对象

2.2 时空对象的表示

时空对象 O_i 可用记录 $(o_id, p_i(t), e_i(t), t)$ 来表示^[1]。其中 o_id 是对象标识符,它唯一标识每个对象 O_i ; $p_i(t)$ 和 $e_i(t)$ 分别对应于对象 O_i 在时间 t 的位置和范围。在各种时空数据

^{*} 国家自然科学基金资助项目(69973022)。黄曙荣 硕士研究生,研究方向为时空对象索引技术。秦小麟 教授,博士生导师,当前研究兴趣为空间分析数据库、时空数据库、GIS等。

库的应用中一般允许在任意时间戳插入和/或删除对象,对象的生命期(lifetime)从对象插入时开始到被删除时结束,在生命期内的对象称为活动的(alive),否则为不活动的(non-alive)。时空数据库在时间戳 t 的状态 $S(t)$ 包括所有在时间戳 t 活动的对象,可将 $S(t)$ 看成是时空对象在时间戳 t 的空间位置和范围的快照。因为时空数据库存储的是随时间变化的空间对象,理论上讲它保存了所有的时空状态 $S(t)$,并可对任意时间戳的 $S(t)$ 进行查询,这意味着时空数据库中时空对象数据的删除仅是逻辑删除,即已删除对象的记录仍保存在数据库中(成为不活动的对象)。

2.3 时空对象的时间维特点

时空数据库可看成是空间数据库在时间维上的扩展,在分析时空数据时应考虑到其时间维的特点。

(1) 时间信息的变化总是单调递增的。

(2) 在时间数据库中,定义了两个时间维度:有效时间和事务时间。有效时间是指现实世界中事件发生的时间,事务时间指信息存储到数据库中的时间。支持事务时间数据库的特点是各种变化的发生时间是按递增顺序的,且通常是对数据库的当前状态操作,这种数据库记录了对象变化的整个过程(历史)。时空数据库也应支持有效时间、事务时间或双时间(bitemporal)。

(3) 数据库的主要目标是准确反映现实世界,空间对象随时间的递增,其变化频率的快慢决定了数据库表示时空数据的方式,可分为离散(discrete)和连续(continuous)两种情况。离散情况下,空间对象的变化频率慢,可用时空状态 $S(t)$ 的快照(snapshot)集表示时空数据,在时空对象发生变化时更新数据;连续情况下的空间对象变化频率快,其空间位置和范围随时间的变化情况可分别表示为时间函数 $p_i(t)$ 、 $e_i(t)$,且只在对象变化的特征(如运动的速度、方向等)改变时才更新数据。

3 时空索引的分类

参照 Y. Theodoridis 等在文[2]中对时空索引的描述,结合目前在时空索引方面已取得的研究成果,时空索引可作如下分类:

(1) 按处理的数据类型分类 主要的时空数据类型包括点数据和区域数据。一部分索引只处理点数据^[3,4]或移动点的轨迹,支持轨迹查询^[5~9];处理区域数据的存取方法既能处理点数据又能处理非点数据,因为点对象可看成是大小为零的区域对象,如3D R-树、RT-树、HR-树、PPR-树及MV3R-树等^[10~14]。

(2) 按数据加载的方式分类 空间索引根据索引创建时数据是以批处理方式全部载入还是动态插入/删除生成索引结构分为静态的(static)和动态的(dynamic)两类,但在时空数据库中,时间维的存在意味着所有对过去时间戳的空间数据插入或更新均是无意义的(无论是静态载入还是动态插入)。因此,时空索引可分为静态的、按时间顺序的(chronological)和动态的三种类型。静态类型和按时间顺序类型均不允许对过去时间戳的数据插入或更新,前者是以批处理方式载入数据(如3D R-树),后者按时间戳的递增顺序动态插入数据(RT-树、HR-树、PPR-树等);动态类型允许动态插入数据,且可对过去时间戳的数据插入或更新。

(3) 按数据集的更新情况分类 根据空间对象是静止/移

动及时间是静态/动态的不同,时空索引可按数据集的更新情况分为增长的(growing)、变化的(evolution)和完全动态的(full-dynamic)三类。增长类型是指对象静止不动,而时间是动态递增的^[10];时间是静态的,而对象是移动的,这样的索引属变化类型;支持时间动态变化情况下的移动对象称为完全动态类型(full-dynamic),目前研究的索引结构大多是支持时间动态变化的^[11~14]。

(4) 按时空对象的近似表示分类 部分索引结构(如3D R-树)采用R-树中用最小边界矩形(MBRs)表示空间对象的方法表示时空对象。由于时空对象独有的移动特性,这种表示方法会导致大量的无效空间和MBRs的重叠,并不能很好地支持时空查询,文[7,12,13]中各自提出了不同的时空对象近似表示方法。

(5) 按是否支持特定的时空查询分类 索引的目的是有效支持用户定义的各种查询操作。基本的空间查询包括选择查询(selection queries)、连接查询(join queries)和最近邻居查询(nearest-neighbor queries)。时空索引除了要支持这三种基本查询外,还应支持时间查询和复杂的时空查询,如时间片查询(timeslice queries)(时间片指特定的时间戳或时间间隔)、轨迹查询(trajecory queries)以及历史查询(history queries)等:RT-树索引结构仅支持基本的时空查询,HR-树、PPR-树支持对时间戳的有效查询,3D R-树和MV3R-树可同时支持时间戳、时间间隔的查询;STR-树^[8]、TB-树^[9]以及TPR-树^[7]可支持对移动点的轨迹查询。

(6) 按支持的时间维度分类 时空数据库应至少包含一维时间信息,时空索引可分为支持有效时间、事务时间和双时间的索引机制。目前大多数的时空索引均支持事务时间,文[15]中提出了支持双时间查询的索引结构R"-树。

(7) 按离散/连续的情况分类 目前大多数的索引结构均是基于离散情况提出的(如3D R-树、HR-树、RT-树以及PPR-树等),可以查询时空对象过去及当前的状态;连续情况下可根据时空对象位置和/或范围随时间变化的函数建立索引,不仅可以查询对象过去、当前的位置和/或范围,还可根据对象当前的信息推测出它在将来某个时间戳的行为,文[5~9]讨论了支持连续情况下查询的索引结构。

4 离散情况下的时空索引机制

离散情况下的时空存取方法(STAM)主要分为以下三类:

(1) 将时间维看成是另一个空间维度

将时间看成是时空对象另一维度的空间信息,这样不仅简单,且可直接使用传统的空间存取方法(如R-树、四叉树)处理 $d+1$ 维的空间信息(d 是参考空间的维度)。文[10]提出的3D R-树将二维参考空间中的最小边界矩形转换成三维空间的最小边界矩形(MBRs的高度对应于对象的生命期),以表示位置及范围均不随时间发生变化的时空对象,并使用R-树索引结构存储时空数据。图2给出了时空对象实例及其对应的3D R-树。

3D R-树实现较容易,可有效地支持时间片查询,缺点是没有考虑到时间维的特有属性,生命期长或/和位置变化大的对象,都会对应较长/大的三维MBRs,引起3D R-树中结点MBRs的大量重叠,会降低查询的效率。

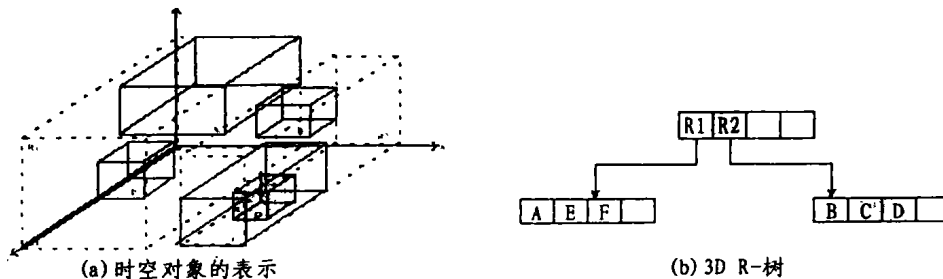


图2 用R-树结构存储三维的MBRs

(2) 对象的时间信息与空间信息一起保存

以MR-树和RT-树^[11]为代表的存取方法是将时间信息和空间信息一起存储到R-树结点中,无论是叶结点还是非叶结点中的实体,都以 (S, T, P) 格式存储,其中 S 表示空间信息(MBRs), T 是时间信息(时间间隔), P 为指向子树或具体对象的指针。这里 $T = (t_i, t_{i+1})$, $i \leq j$, t_i 表示当前时间戳, t_{i+1} 是下一个时间戳($t_i \leq t_{i+1}$),如果对象在时间戳 t_i 到 t_{i+1} 期间空间位置不变,则实体的空间信息 S 保持不变,而时间信息由 T 更新为 T' ($T' = (t_i, t_{i+1})$);一旦对象的空间位置发生变化,则原实体保持不变,产生新的实体 (S', T', P') , (S' 是对象新的空间位置, $T' = (t_{i+1}, t_{i+2})$)插入到RT-树中。

这种结构较适用于对象移动较少的时空数据库,如果数据库中对象的移动量较大,实体数目的增长会很可观。另外,每个实体中均包含对象实例的空间信息和时间信息,插入和分裂策略很难同时满足对对象的空间和时间属性的查询要求(如插入溢出时按哪个属性分裂结点)。

(3) 使用部分持久和重叠技术存储不同时间戳时空状态的方法

部分持久结构^[16](partially persistent structure)将二维

空间存取方法(如R-树、四叉树或k-d树结构)转换为部分持久的,它在“逻辑”上存储过去的时空状态 $S(t)$,并只对当前时空状态数据更新。对时空状态 $S(t)$ 的历史查询可直接对时空状态在时间戳 t 的部分持久结构操作(可看成是对暂态R-树的操作),返回结构中的在时间戳 t 的活动对象。该结构避免了3D R-树中长生期对象引起的结点MBRs重叠问题,可有效地支持时间戳和时间间隔查询。

重叠(overlapping)技术的思想是:用二维空间存取方法为每个时间戳建立一棵对应的暂态树(如暂态R-树),并假设相邻时间戳的暂态树的区别不大,存在许多共同路径(路径上结点内空间对象没有随时间变化)。为了节省磁盘空间,相邻时间戳的暂态树间可共享这些路径,而那些包含空间对象变化的结点及路径可更新后复制到后续暂态树中。如图3所示,图3(a)、(b)分别对应时间戳 t_0 和 t_1 的暂态R-树,如果在时间戳 t_1 结点3数据更新(假设是结点3中一个空间对象被删除),则时间戳 t_1 的暂态R-树与 t_0 时的区别仅是创建了新的结点3a和到达结点3a的新路径,其余结点和路径均与 t_0 时相同,可共享,两棵暂态R-树有重叠部分,见图3(c)。

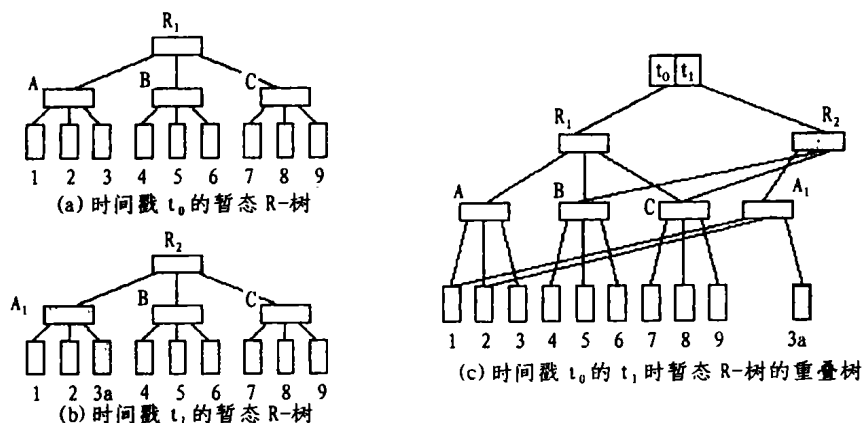


图3 暂态R-树及重叠树

采用重叠的方法操作简单,可适用于多种空间存取方法(如R-树、四叉树等)。文[12~14]提出了各种重叠的R-树,其中HR-树是最有代表性的索引结构;在文[17,18]中讨论了用重叠的线性四叉树存储随时间变化的光栅数据。重叠的方法在对象变化频繁的时空数据中对存储空间的要求较高(最终退化为相互独立的暂态树)。

文[12]提出的PPR-树更好地结合了部分持久结构和重叠的方法,对长生命期时空对象的MBRs进行了分裂优化(减少无效空间),可支持快速查询,减少对存储空间的要求。

5 连续情况下的时空索引机制

连续情况下的研究热点是如何对连续移动点的轨迹索引。

(1) 移动点过去轨迹的索引方法

对连续移动点过去轨迹的索引一般可转化为离散情况下的索引问题。例如,在平面上作直线运动的点对象,其运动轨迹可看成是若干线段的集合。对象的移动变化意味着在原来轨迹的末端插入新的线段。如果将每条线段都当作是一个空间对象(如点或区域对象)处理,则连续点的移动变化相当于

在数据库中插入和删除线段对象,这正是在离散情况下时空对象移动的特殊问题(即只有对象的插入和删除,对象的位置和范围不作变化),可对离散情况下的时空存取方法进行扩展后创建索引,如 STR-树、TB-树。

(2) 支持移动点未来位置查询的索引方法

连续情况下移动点的运动轨迹可用时间函数(如 $p_i(t)$)表示,根据 $p_i(t)$ 及已知的某一时间戳移动点的位置来预测其未来位置。

创建索引的一种简单方法是假设点对象在 d -维($d=1, 2, 3$)空间移动,这些点未来的移动轨迹可当作 $d+1$ -维空间的线对象进行索引。如 J. Tayeb 等将一维空间中点的移动轨迹看成是 (x, t) 空间中的线,使用 PMR-四叉树(PMR-

quadtree)作为索引结构^[6]。具体方法是将时间维划分为若干个相等的时间间隔 ΔT ,对每个 ΔT 重建新的 PMR-树索引,并允许在 ΔT 内插入、删除和更新操作。这种索引结构的缺点是在树中存在许多具体数据的复制(如一条线段通常会存储在几个结点中),且需要大量的存储空间和频繁的数据更新。

另一种建索引的方法是 G. Kollios 等提出的在二元空间中建索引的方法^[5],其主要思想源于几何算法中的二元转换:将原来空间(primal space)中的线(移动点的轨迹)转换为二元空间(dual space)中的点,用点存取方法(PAM)建索引,如图4所示。实验表明^[5],在二元空间中,基于 k -d-树的存取方法优于 R-树。

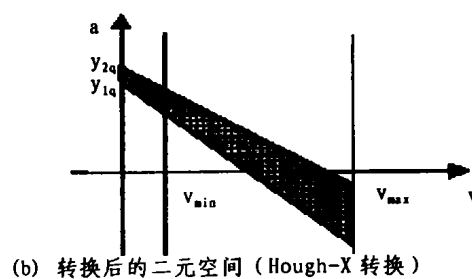
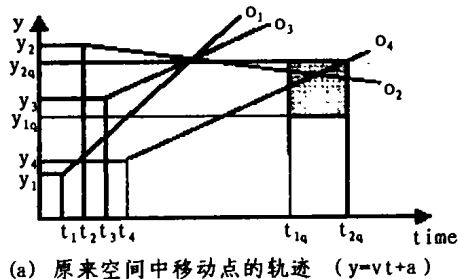


图4 二元空间转换

文[7]提出了直接在 d -维空间对移动点建索引的结构-TPR-树(time-parameterized R-tree):用速度矢量和 MBRs 两个参数表示移动对象,其中 MBRs 是以时间为参数的最小边界矩形,即它包围的是结点中当前时间的所有实体。在响应

对未来任一时间戳的查询时,先计算这个未来时间戳的 MBRs,在与查询窗口比较后返回相交区域的所有活动对象,如图5所示。

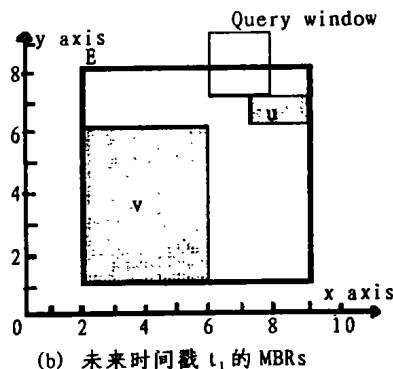
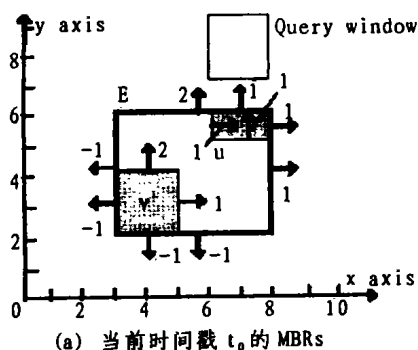


图5 TPR-树中的 MBRs

6 我们的工作

我们目前正在研制时空分析数据库管理系统 STADBS,采用将时空子系统 ROSTE (Robust Spatial-Temporal Extension)集成到可扩展数据库系统中的开发策略^[19]。时空子系统 ROSTE 基于 Realms^[20]和主存技术,具有支持时空对象的表示、存储、管理、分析操作(尤其是拓扑分析)和索引机制等功能。

我们现已设计和实现了 ROSTE 中的时空对象存储管理子系统^[21]及支持时空对象的空间分析操作算法^[22]。在此基础上,正在设计和实现采用 PPR-树作为 ROSTE 的索引结构,借鉴 G. Kollios 在文[13]中提出的 MBRs 优化分裂算法,研制基于 Realms 和主存的支持时空对象高效存取、查询的索引机制,以保障 STADBS 强大的时空分析功能的效率。

结束语 时空索引机制的研究工作现仍处于探索阶段,

参考文献

- 1 Manolopoulos Y, Theodoridis Y, Tsotras V J. Advanced Database Indexing. Kluwer Academic Publishers, 1999
- 2 Theodoridis Y, Sellis T, Papadopoulos A, Manolopoulos Y. Specifications for Efficient Indexing in Spatiotemporal Databases. STDBM, 1998
- 3 Agarwal P, Arge L, Erickson J. Indexing Moving Points. ACM PODS, 2000
- 4 Nascimento M, Silva J, Theodoridis Y. Evaluation of Access Structures for Discretely Moving Points. STDBM, 1999

- 5 Kollios G, Gunopulos D, Tsotras V J. On Indexing Mobile Objects. In: Proc. of the PODS Conf., 1999. 261~272
- 6 Tayeb J, Ulusoy O, Wolfson O. A Quadtree Based Dynamic Attribute Indexing Method. The Computer Journal, 1998, 41(3): 185~200
- 7 Saltenis S, Jensen C S, Leutenegger S T, Lopez M A. Indexing the positions of continuously moving objects. In: Proc. ACM SIGMOD Intl. Conf. on Management of Data, 2000. 331~342
- 8 Pfoer D, Theodoridis Y, Jensen C S. Indexing Trajectories of Moving Point Objects: [Chorochronos Tech. Rep. CH-99-3]. June 1999
- 9 Pfoer D, Jensen C, Theodoridis Y. Novel Approaches to the Indexing of Moving Object Trajectories. VLDB, 2000
- 10 Theodoridis Y, Vazirgiannis M, Sellis T. Spatio-Temporal Indexing for Large Multimediapplications Proceedings of the 3rd IEEE Conference on Multimedia Computing and Systems (ICMCS), 1996
- 11 Xu X, Han J, Lu W. RT-tree: An Improved R-tree Index Structure for Spatiotemporal Databases. In: Proc. of the 4th Intl. Symposium on Spatial Data Handling (SDH), 1990
- 12 Nascimento M, Silva J. Towards Historical R-trees. ACM SAC, 1998
- 13 Hadjieleftheriou M, Kollios G, Tsotras V J, Gunopulos D. Efficient indexing of spatiotemporal objects. In: Proc. VIII Conf. Extending Database Technology, 2002. 251~268
- 14 Tao Y, Papadias D. The MV3R-Tree: A Spatio-Temporal Access Method for Timestamp and Interval Queries. In: Proc. of 27th Very Large Data Bases Conf. (VLDB), 2001
- 15 Saltenis S, Jensen C S. R-tree Based Indexing of General Spatio-Temporal Data: [TimeCenter Technical Report]. 1999
- 16 Kumar A, Tsotras V J, Faloutsos C. Designing Access Methods for Bitemporal Databases. IEEE TKDE, 1998, 10(1): 1~20
- 17 Tzouramanis T, Vassilakopoulos M, Manolopoulos Y. Overlapping Linear Quadrees: a Spatio-Temporal Access Method. In: Proc. of the 6th ACM Intl. Workshop on Advances in Geographic Information Systems, 1998. 1~7
- 18 Tayeb J, Ulusoy O, Wolfson O. A Quadtree-Based Dynamic Attribute Indexing Method. The Computer Journal, 1998, 41(3): 185~2008
- 19 秦小麟. 空间分析数据库的研究方法及技术. 中国图象图形学报, 2000, 5(9)
- 20 Güting R H, Schneider M. Realms: A Foundation for Spatial Data Types in Database Systems. In: Proc. 3rd Intl. Symposium on Large Spatial Databases, Singapore, June 1993. 14~35
- 21 郑翰. 时空分析 DBMS——STADBS 的数据模型与存储机制研究: [硕士学位论文]. 南京航空航天大学, 2002. 3
- 22 邹永娟. 时空分析数据库 STADBS 的分析操作算法及查询语言: [硕士学位论文]. 南京航空航天大学, 2002. 3

(上接第62页)

源 IP 地址--当前使用的 IP 地址 (支持 IP6 地址)

目的 IP 地址--当前访问的 IP 地址 (支持 IP6 地址)

当前时刻--当前的绝对时间

数据流量--以字节为单位, 是一个绝对数字, 单调递增

应用类型--包括电话、电视、数据服务等

业务类型--该业务的服务质量要求, 包括如下内容:

服务等级--该业务要求的服务级别

要求的最大带宽--该业务要求的最大带宽

要求的平均带宽--该业务要求的平均带宽

时延要求--该业务要求的最大允许延迟

服务等级包括:

普通业务: 仅计算相关的流量, 最大带宽、平均带宽和时延要求等字段都无效。(等级1)

平均带宽要求: 仅要求平均带宽(等级2)

最大带宽要求: 仅要求平均带宽和最大带宽(等级3)

时延要求: 要求平均带宽、最大带宽和时延(等级4)

·业务类型参数的具体描述如下:

业务类型号--每个业务类型的标示

服务等级号--前述的4种服务等级

应用类型--包括电视、电话、普通数据服务等

最大时延范围--以毫秒为单位

最大带宽范围--以 bps 为单位

平均带宽范围

时段--本业务类型适用的时间范围

源地址范围--本业务类型的使用用户的地址范围

目的地址范围--本业务类型适用的目的地址范围

时长费率--用户使用网络单位时间的费用

流量费率--用户得到数据流量的费用计算单位

调节常数--可以理解为使用此业务类型的基础费用

·用户组信息的具体描述如下:

用户组名

总折扣率--对本用户组所有业务的统一折扣率

最低消费值--加入本用户组的用户每月的最低消费值

业务类型号--前面定义的各种业务类型

折扣率--本用户组中每种业务类型对应的费率

结束语 随着网络的迅速发展, 各种新兴业务不断出现, 实现 QoS 将成为未来网络服务的主流, 基于流量和时间的计费模式会逐渐显现出它的缺点, 将会逐渐被基于 QoS 的网络计费所替代。网络 QoS 服务还处于研究开发阶段, 相应地基于 QoS 的网络管理功能也有许多需要解决和继续深入探讨的问题。本文所述基于 QoS 的计费策略是在解决实际问题的基础上提出的, 具有一定的精确性和公平性, 可以作为一个有益的思路继续研究下去。

参 考 文 献

- 1 Hapnedy S. 简单网络管理协议教程(第2版). 电子工业出版社
- 2 Braden R, Clark D, Shenker S. Integrated Services in the Internet Architecture: an Overview. RFC1633 <http://www.ietf.org/rfc/rfc1633.txt>
- 3 Shenker S, Wroclawski J. Network Element Service Specification Template. RFC2216 <http://www.ietf.org/rfc/rfc2216.txt>
- 4 Braden R, Zhang L, Berson S, Herzog S, Jamin S. Resource Reservation Protocol (RSVP). RFC2205 <http://www.ietf.org/rfc/rfc2205.txt>
- 5 Blake S, Black D, Carlson M, Davies E, Wang Z, Weiss W. An Architecture for Differentiated Services. RFC2475 <http://www.ietf.org/rfc/rfc2475.txt>