

P2P 数据管理系统中数据表的定位

申新鹏¹ 李战怀² 赵晓南² 曾雷杰²

(西安工业大学计算机科学与工程学院 西安 710032)¹ (西北工业大学计算机学院 西安 710072)²

摘要 P2P 数据管理系统已经成为对等计算领域的研究重点。语义异构是 P2P 数据管理系统的首要问题。为了解决此问题,在每个数据源节点对共享的数据表的表名和属性名分别定义一系列关键字作为语义映射的媒介,具有相同关键字的异构数据源之间自动建立映射关系。这些关键字就形成了共享数据的外模式。但在节点内部,没有将外模式真正地物化为视图。定义好的关键字使用外模式描述文件分布到整个网络中。在查询的过程中,找到外模式描述文件后,立即将查询请求中的所有别名转换为真实的数据表名和属性名,从而既可以方便地按照任意名字找到需要的数据表,又可以减少数据备份的数量,简化查询算法,提高系统效率。

关键词 对等计算,数据管理,语义异构,数据缓存,数据表定位

中图法分类号 TP393 **文献标识码** A

Locating Table in P2P Data Management System

SHEN Xin-peng¹ LI Zhan-huai² ZHAO Xiao-nan² ZENG Lei-jie²

(Computer Science and Engineering College, Xi'an Technological University, Xi'an 710032, China)¹

(College of Computer, Northwestern Polytechnical University, Xi'an 710072, China)²

Abstract The data management system has become the focus of peer-to-peer computing. Semantic heterogeneity is the first problem to P2P data management system. In order to solve this problem, in each data source node, some keywords of table names and attribute names were defined as the medium of semantic mapping. The semantic mapping was automatically created between the heterogeneous data source nodes, which had the same keywords. The keywords became the external schema of the shared tables. But the keywords were not really materialized to the views. All the keywords were distributed to the P2P network using the external schema description files. When a query was executed, once the external schema description files were found, all the aliases would be immediately changed into real table names or attribute names. Using the way, the amount of data backup was reduced, the search algorithm was simplified and the system efficiency was improved.

Keywords Peer-to-peer computing, Data management, Semantic heterogeneity, Data caching, Locating data table

1 问题描述

基于对等计算的文件共享系统取得了很大的成功。但是在这种文件共享系统中,用户只能通过文件的标识符进行查找,有时不能取得满意的效果。于是结构化数据被引入到了对等计算中,产生了 P2P 数据管理系统^[1,2]。

P2P 数据管理系统是一个多数据库系统,系统中有多个数据源,但是不同的数据源对相同的现实世界的实体可能会使用不同的描述,有不同的表示。这样必然会产生语义异构的问题。

传统的信息集成领域使用集中式的策略解决语义异构问题:采用中介系统,将全局查询分解后发给各数据源;或采用物化的途径,将所有数据源集合到一个中心位置;或使用统一的全局模式管理所有的数据。

P2P 数据管理系统是基于动态广域网的,不存在中心控制结点,无法形成统一的全局视图。要在这样的环境下有效地解决语义异构问题,实现数据整合,就必须在不同数据源之间建立语义映射。

语义映射是以单个的数据表为单位的。但一次查询往往同时涉及到一个数据库中的多个数据表,且必须同时找到所有需要的数据表才能进行查询。在进行多表查询定位的过程中,还必须将其他数据库的同名数据表去除。由此可见,进行多表定位有很多问题需要讨论解决。

作者对此进行了深入的研究,提出了一种分层索引机制来解决这个问题。配合作者提出的多节点集群 P2P 系统^[3]和基于距离的数据表查询算法^[4],可以在 P2P 数据管理系统中实现单表或多表的精确匹配查询和范围查询。

分层索引机制就是将数据库的定位及数据表的查询和元

到稿日期:2010-04-21 返修日期:2010-07-12

申新鹏(1975-),男,博士生,讲师,主要研究方向为网络计算、数据管理等,E-mail:shenxinpeng@163.com;李战怀(1961-),男,教授,博士生导师,主要研究方向为存储区域网络、数据库系统等;赵晓南(1979-),女,博士生,讲师,主要研究方向为海量数据存储、数据管理;曾雷杰(1978-),男,博士生,讲师,主要研究方向为海量数据存储、数据管理。

组的发现相分离,分别实现。本文主要关注数据库的定位及数据表的查询问题。

2 数据表在 P2P 网络中的存储

2.1 网络结构

针对 P2P 数据管理系统的特点,作者设计了一个多节点集群 P2P 系统^[3]。在这个系统中,多个节点聚集在一起形成一个集群。集群作为一个整体参与 Chord^[5] 环的形成,如图 1 所示。

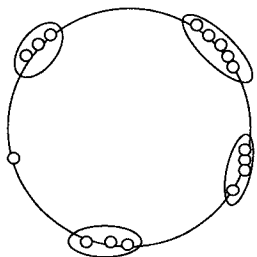


图 1 多节点集群 P2P 系统

集群内部多个节点可以根据需要组成小的子系统。如果集群中节点很少,可以两两相互连接形成网络。如果集群中节点很多并且需要数据共享,可以使用任意 P2P 算法形成 P2P 子网。

在生成多节点集群 P2P 网络系统时,可以将相同的数据在集群中增加多个备份,以提高数据的安全性和下载速度。另一方面,也可以将语义相关的数据分配到相同的集群中,从而实现语义聚集。

本文以下的设计都是基于多节点集群 P2P 网络的。

2.2 数据资源缓存

P2P 数据管理系统中有多个节点共享其数据库。缓存同一数据库的多个节点聚集在一起形成一个集群。这样在执行查询的过程中,找到一个节点之后就可以方便地找到其他节点,简化查询算法。针对同一数据库中多个数据表的缓存在集群子网中的分布方法参见文献[4]。缓存子网可以使用任意一种结构化 P2P 算法实现。使用 Chord 协议实现的集群子网如图 2 所示。

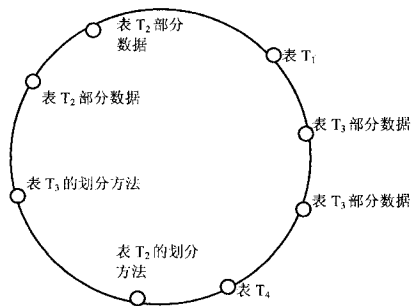


图 2 缓存数据源节点子网

3 语义异构问题

P2P 数据管理系统中有多个数据源,不同的数据源对相同的现实世界的实体可能会使用不同的描述,有不同的表示。这样必然会产生语义异构的问题。

为了解决语义异构问题,在每个数据源节点对共享的数据表的表名和属性名分别定义一系列关键字作为语义映射的

媒介。具有相同关键字的异构数据源之间自动建立映射关系。这些同义关键字就形成了共享数据的外模式。通过使用外模式,原来共享的一组数据,现在看来变成了多组具有相似名称的数据。在节点内部,并没有将外模式真正地物化为视图,而只是建立了外模式和内模式(数据表的真实模式)之间的对应关系。在进行查询时,使用后面介绍的算法,先将基于外模式的查询条件转换为基于内模式的查询条件,然后再实际执行。

外模式以描述文件的形式存放在节点中。外模式描述文件的产生和维护由数据源节点管理员通过半自动的方式完成。如果数据表的缓存由一个节点保存(见图 2 中的表 T_1),外模式描述文件由该节点负责保存和推送。如果数据表的缓存由多个节点保存(图 2 中的表 T_2),外模式描述文件由保存表的划分方法的节点负责保存和推送。

外模式描述文件除了记录表及属性的别名,还记录了推送描述文件的节点 IP 及数据表所在的集群标识符。外模式描述文件的例子如图 3 所示。

```
<? xml version="1.0"?>
<Catalog>
  <Cluster ID=101110>
    <Node IP=10.25.63.215>
      <DataTable>
        <Tablename>学生信息表,同学信息表</Tablename>
        <Attribute>学号,学生序号,StudentID</Attribute>
        <Attribute>姓名,学生名字,Name</Attribute>
        <Attribute>身高,学生身高,Height</Attribute>
        <Attribute>年龄,Age</Attribute>
      </DataTable>
    </Catalog>
```

图 3 外模式描述文件示例

有了外模式描述文件之后,还需要将它推送到整个 P2P 网络中。由保存外模式描述文件的节点分别按数据表的所有别名经哈希^[6]运算得到文件标识符(如图 3 所示的外模式描述文件,分别按表名“学生信息表”和“同学信息表”进行哈希运算得到文件标识符)。然后将外模式描述文件发送到对应的后继集群中。后继集群收到外模式描述文件后,将其保存到集群中一个或多个节点中供查询时使用。

不同数据库中会有相同名字的数据表,外模式描述文件是根据表名哈希运算得到的文件标识符并推送到 P2P 网络中的。因此这些来自于不同数据库的同名数据表的外模式描述文件就聚集在同一个集群中了。

4 数据表的定位

数据表的定位是以外模式描述文件为基础进行的。根据查询涉及到的数据表的数量,可以分为单表查询和多表查询。

4.1 单表查询时数据表的定位

外模式描述文件是根据数据表的名字作为关键字经过哈希运算分散到 P2P 网络中的。如果一个数据表在外模式描述文件中有 K 个别名,则需要将外模式描述文件分别以每个别名为关键字进行哈希运算后分散到网络中,共有 K 个备份。

在外模式描述文件中,每个数据表有多个别名,同时数据表中的每个属性也有多个别名。如果所有可能的表名-属性名组合都对应一个完整的数据备份,将会产生大量的冗余数

据,增加系统的存储开销和更新难度。为了降低系统复杂度,提高查询效率,可以根据外模式描述文件对查询条件进行修改,从而达到减少备份数量,提高数据资源利用率的目的。

例如图3所示的外模式描述文件。其中关系表的真实数据模式为:学生信息表(学号,姓名,身高,年龄)。但该表还有一个别名,4个属性也分别有各自的别名。如果对所有可能的组合都对应一个数据备份,这个表需要 $2 * 3 * 3 * 3 * 2 = 108$ 个备份,这显然是不可能的。因此,在找到外模式描述文件之后,可以对查询条件进行转换:将其中所有的别名替换为数据表中真实的名字。在后续的查询过程中都使用替换过的查询条件。

例如对如下查询请求:

```
select 学生序号,姓名,学生身高, Age from 同学信息表  
可以替换为:
```

```
select 学号,姓名,身高,年龄 from 学生信息表
```

从以上的描述可以看出,进行数据表的查询需要考虑相关的诸多因素。通过对前面提到的这个查询例子的详细介绍来演示数据表查询的全过程。

1. 以“同学信息表”为关键字进行哈希运算得到文件标识符,然后在P2P网络中找到后继集群。

2. 在后继集群中如果没有需要的外模式描述文件,则查询失败,退出。如果有,转下一步。

3. 分别打开找到的外模式描述文件,并查找是否有表名为“同学信息表”,并且含有“学生序号,姓名,学生身高, Age”属性的数据表,如果没有,则查询失败,退出。如果找到,转下一步。

4. 进行查询条件的转换。将查询条件中的表名替换为对应的<Tablename>后的第一个表名,将查询条件中所有的属性名也替换为对应的<Attribute>后的第一个属性名。

5. 在外模式描述文件中有IP地址,这是推送外模式描述文件的节点的IP地址,该节点也必然处于共享该数据表的集群中。与该节点建立连接,如果连接成功,就进入元组查询阶段。如果连接不成功,转下一步。

6. 如果连接不成功,可能是该节点失效造成的,但节点失效,共享数据表的集群可能仍然是有效的。读取外模式描述文件中的集群标识符(Cluster ID),根据Cluster ID查找共享该数据表的集群。如果找到就进入元组查询阶段。如果找不到,异常退出。

4.2 多表查询时数据表的定位

多表查询是多个数据表通过一个或多个属性进行连接而进行的查询。最简单的多表连接查询是两个数据表通过一个属性进行连接。例如有两个数据表R,S,通过属性R.a和S.b进行连接查询如下:

```
Select * From R,S Where R.a=S.b
```

多表查询同时涉及到多个数据表,因此必须确定查询涉及到的多个数据表和属性都在同一个数据库系统中,才能进行数据查询。但是外模式描述文件是以数据表为单位分散到整个网络中的,确定一个数据表所在的集群并不能确定多表连接查询可以被执行。

为此需要对查询过程稍加修改。先使用前面介绍的方法确定第一个表所在的数据表,并将查询请求发送到数据库所在的集群。然后判断查询涉及到的其它的数据表是否在集群

中。具体来说,就是以数据表的名字作为关键字在集群子网内查询,如果找到对应的元数据外模式描述文件并且需要的属性也都存在,则继续进行元组查询,否则,查询失败。

5 相关工作

针对语义异构和数据表定位问题,不同的研究者提出了一些不同的解决方案。

PeerDB^[7]项目使用关键字标注关系名和属性名的方法实现异构数据源的语义映射。同时对每一组关键字都建立了相应的物化视图。但PeerDB项目是以非结构化P2P网络为基础的,同一数据库的多个数据表不能形成语义聚集,无法解决多表连接查询的问题。另一方面,对每一组关键字都建立了相应的物化视图,提高了查询效率,但同时增加了数据冗余,降低了系统的灵活性。

Hyperion^[8]项目通过定义复杂的映射表、映射表达式和ECA^[9]规则控制节点间的信息交换,高效地解决了P2P环境下的语义异构问题。但是这些工作必须由领域专家在节点加入网络时且查询发起前预先建立这些对应关系,因此系统成本较高。

PIER^[10]项目假定系统中所有的节点都服从某个皆知的标准全局模式,没有定义专门的算法来解决节点间语义异构问题。

结束语 P2P数据管理系统中节点的自治性造成了不同数据源的数据模式的异构性,因此必须进行数据整合,为用户提供一个统一的查询接口。

本文在每个数据源节点对共享的数据表的表名和属性名分别定义一系列关键字作为语义映射的媒介。具有相同关键字的异构数据源之间自动建立映射关系。这些同义关键字就形成了共享数据的外模式。使用这种方法,原来共享的一组数据,现在看来变成了多组具有相似名称的数据。但在节点内部,并没有将外模式真正地物化为视图,而只是建立了外模式和内模式之间的对应关系。

定义好的关键字列表使用外模式描述文件分布到整个网络中,在生成外模式描述文件时,包含数据表和属性的所有别名,并将外模式描述文件按所有数据表的别名分散到P2P网络中。但在查询的过程中,找到外模式描述文件后,又立即将查询请求中的所有别名转换为真实的数据表名和属性名,从而既可以方便地按照任意名字找到需要的数据表,又可以减少数据备份的数量,简化查询算法,提高系统效率。

参考文献

- [1] 余敏,李战怀,张龙波. P2P数据管理[J]. 软件学报,2006,17(8):1717-1730
- [2] Huebsch R, Chun B, Hellerstein J, et al. The architecture of PIER: An Internet-scale query processor[C]//Proc. of the 2005 Conf. on Innovative Data Systems Research. Asilomar, VLDB, 2005:28-43
- [3] 申新鹏,李战怀. 多节点集群P2P系统研究[J]. 计算机科学,2010,27(2):71-74,93
- [4] Shen Xin-peng, Li Zhan-huai. Multi-dimensional Queries in DHT-based Peer-to-Peer Systems[C]//Proceeding of 5nd International Conference on Semantics, Knowledge and Grid (SKG 2009). Zhuhai, China, IEEE CS, 2009:60-65

[5] Stoica I, Morris R, et al. Chord A Scalable Peer-to-peer Lookup Protocol for Internet Applications[J]. IEEE/ACM Transactions on Networking, 2003, 11(1): 17-32

[6] Karger, Lehman, et al. Consistent hashing and random trees Distributed caching protocols for relieving hot spots on the World Wide Web[C]//Proceedings of the 29th Annual ACM Symposium on Theory of Computing. El Paso, TX: ACM Press, 1997: 654-663

[7] Ng W S, Qoi B C, et al. PeerDB A P2P-based System for Distributed Data Sharing[C]//Proc. of the 19th ICDE. Bangalore: IEEE Computer Society Press, 2003: 633-644

[8] Kementsietsidis, Arenas M. Data sharing through query translation in autonomous sources[C]//Proc. of the 30th Int'l Conf. on Very Large Data Bases. San Francisco: Morgan Kaufmann Publishers, 2004: 468-479

[9] Zhao Dan, Mylopoulos J, et al. An ECA Rule Rewriting Mechanism for Peer Data Management[C]//Proceedings of the International Conference on Extending Database Technology (EDBT). Munich, Germany: Springer Berlin/Heidelberg, 2006: 1069-1078

[10] Huebsch R, Chun B, et al. The Architecture of PIER: an internet-scale query processor[C]//Proc. of the 2005 Conference on Innovative Data Systems Research. Asilomar: Online Proceedings, 2005: 28-43

(上接第 174 页)

表 3 实体 c 的直接信任记录

服务	需求上下文	信任
W1	E1= $\langle\langle 4,6 \rangle, \langle 0.6,0.9 \rangle, \langle 0.8,0.95 \rangle, \langle 30,40 \rangle\rangle$	0.52
W1	E2= $\langle\langle 3,5 \rangle, \langle 0.65,0.85 \rangle, \langle 0.6,0.9 \rangle, \langle 25,50 \rangle\rangle$	0.6
W1	E3= $\langle\langle 4,5,5 \rangle, \langle 0.6,0.9 \rangle, \langle 0.8,0.9 \rangle, \langle 28,40 \rangle\rangle$	0.7
W1	E4= $\langle\langle 3,5,6 \rangle, \langle 0.5,0.9 \rangle, \langle 0.85,0.95 \rangle, \langle 30,42 \rangle\rangle$	0.67
W1	E5= $\langle\langle 2,5 \rangle, \langle 0.7,0.9 \rangle, \langle 0.55,0.95 \rangle, \langle 50,80 \rangle\rangle$	0.65
W1	E6= $\langle\langle 6,9 \rangle, \langle 0.85,1 \rangle, \langle 0.7,1 \rangle, \langle 10,25 \rangle\rangle$	0.5
W1	E7= $\langle\langle 7,11 \rangle, \langle 0.5,0.9 \rangle, \langle 0.7,0.9 \rangle, \langle 25,50 \rangle\rangle$	0.90
W1	E8= $\langle\langle 6,10 \rangle, \langle 0.4,0.9 \rangle, \langle 0.5,0.9 \rangle, \langle 30,60 \rangle\rangle$	0.95
W2	E1= $\langle\langle 7,11 \rangle, \langle 0.5,0.9 \rangle, \langle 0.7,0.9 \rangle, \langle 25,50 \rangle\rangle$	0.42
W2	E2= $\langle\langle 6,10 \rangle, \langle 0.4,0.9 \rangle, \langle 0.5,0.9 \rangle, \langle 30,60 \rangle\rangle$	0.57
W2	E3= $\langle\langle 6,20 \rangle, \langle 0.2,0.7 \rangle, \langle 0.4,0.85 \rangle, \langle 20,50 \rangle\rangle$	0.83
W2	E4= $\langle\langle 10,20 \rangle, \langle 0.3,0.6 \rangle, \langle 0.5,0.9 \rangle, \langle 40,60 \rangle\rangle$	0.9
W2	E5= $\langle\langle 8,18 \rangle, \langle 0.3,0.8 \rangle, \langle 0.5,0.9 \rangle, \langle 40,65 \rangle\rangle$	0.78
W2	E6= $\langle\langle 9,5,18 \rangle, \langle 0.3,0.7 \rangle, \langle 0.4,0.85 \rangle, \langle 35,60 \rangle\rangle$	0.81
W2	E7= $\langle\langle 10,22 \rangle, \langle 0.2,0.55 \rangle, \langle 0.6,1 \rangle, \langle 40,60 \rangle\rangle$	0.85
W2	E8= $\langle\langle 9,15 \rangle, \langle 0.3,0.6 \rangle, \langle 0.6,0.9 \rangle, \langle 50,70 \rangle\rangle$	0.95

假设 c 发出一个新的信任评估请求,需求上下文为 $E_c = \langle\langle 7,10 \rangle, \langle 0.45,0.9 \rangle, \langle 0.6,0.9 \rangle, \langle 28,55 \rangle\rangle$,则它需要根据本次的需求,对 w_1 和 w_2 进行信任评估。利用传统信任评估方法,计算两个服务的直接信任度分别为: $T_{c \rightarrow w_1}^E = 0.71$, $T_{c \rightarrow w_2}^E = 0.77$ 。而从表 2 我们可以看出实际上 w_1 的服务质量比 w_2 好,两个服务都是可信的服务,所以 w_1 的直接信任度应该高于 w_2 。产生这个结果的原因是,在 c 对 w_2 的交互数据中,需求约束比较弱,使 w_2 虽然服务质量低,但是满意度评价却比较高,从而信任度也比较高。而在与 w_1 的交互中,需求约束比较强,使得满意度评价比较低,从而信任度也比较低。

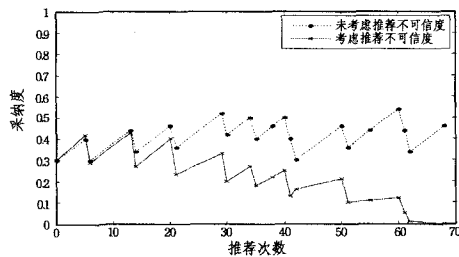


图 2 推荐不可信度对采纳度的影响

采用本文的方法重新对两者的信任度进行计算,得到 $T_{c \rightarrow w_1}^E = 0.79$, $T_{c \rightarrow w_2}^E = 0.41$,可以从评估结果中看出 w_1 的信任度比 w_2 的直接信任度高,这也符合实际情况。

实验二 为了验证推荐不可信度对动态恶意推荐实体的采纳度影响。将传统模型只根据推荐可信度更新采纳度的方

法和本文中根据推荐可信度和推荐不可信度更新采纳度的方法进行了比较。实验设置 $ct_{max} = 10$ 。

从图 2 中可以看出,考虑了推荐不可信度之后,对动态恶意实体起到一定的抑制作用,并且当其不可信推荐的次数达到阈值时,采纳度趋近于 0。

结束语 本文针对分布式环境中 Web 服务选择时的信任评估问题,根据 Web 服务的特点,对已有的信任评估方法进行了改进。在信任评估过程中,考虑了服务质量的需求约束和属性偏好对信任度的影响,并引入推荐不可信度的概念,综合推荐实体的推荐可信度,运用模糊推理的方法计算出推荐实体的采纳度,对具有相同推荐可信度却具有不同推荐能力的推荐实体进行了区分。实验证明了该方法能够得到 Web 服务更加准确可靠的信任信息,为服务请求节点选择满足需求的可信的 Web 服务提供了基础。下一步的研究工作是对实体的直接信任更新引入惩罚和奖励机制,使善意实体能够持续提供高质量可信的服务,甄别出各种恶意实体,减少恶意实体的影响。

参考文献

[1] Beth T, Borcherding M, Klein B. Valuation of trust in open network[C]//Gollmann D, ed. Proc. of the European Symp. on Research in Security(ESORICS). Brighton: Springer-Verlag, 1994: 3-18

[2] Beth T, Borcherding M, Klein B. Valuation of trust in open networks[C]//Proceedings of the European Symposium on Research in security. Brighton: Springer-Verlag, 1999: 59-63

[3] Wang Y, Vassileva J. Bayesian network-based trustmodel[C]//Proceedings of the IEEE/WIC International Conference on Web Intelligence. Halifax, Canada, 2003: 372-378

[4] Jøsang A. An algebra for assessing trust in certificate chains[C]//The Internet Society Symp. on Network and Distributed System Security. San Diego, 1999

[5] Zhou Jue-jia, Mu Chun-di. A kind of application-specific QoS control in wireless sensor networks[C]//IEEE International Conference on Information Acquisition. 2006: 456-461

[6] 高亚春,张为群. 基于 QoS 本体的 Web 服务描述和选择机制[J]. 计算机科学, 2008, 35(12): 273-276

[7] 李春梅,蒋运承. 具有 QoS 约束的语义 Web 服务发现的研究[J]. 计算机科学, 2007, 34(6): 116-121

[8] 陈蜀宇,刘刚国. 面向 Web 服务的数字化营区系统架构[J]. 重庆工学院学报:自然科学版, 2008, 22(9): 103-107