

# 基于 E-R 图的关系数据库关键字查询<sup>①</sup>

李威<sup>1</sup>, 高锦涛<sup>2</sup>, 高腾<sup>2</sup>

<sup>1</sup>(山东建筑大学 计算机科学与技术学院, 济南 250101)

<sup>2</sup>(山东银座购物中心有限公司, 济南 250012)

**摘要:** 随着信息检索技术和数据库理论的发展, 如何通过信息检索技术在关系数据库中获取有用信息, 并使普通用户也可以完成在关系数据库中的关键字查询, 成为近期研究人员研究的热点. 基于关键字的关系数据库查询系统无需用户了解数据库模式和结构化查询语言, 用户只需输入几个关键字就可以得到查询结果. 基于关系数据库的原型 E-R 图, 根据意义的不同对实体进行分组. 查询时根据关键字所属关系表, 确定查询的实体集合, 划定查询范围, 提高查询效率. 提出了新的算法来提高系统的查准率. 最后, 通过实验对 E-RKS 系统效能进行验证.

**关键词:** 关系数据库; 关键字查询; E-R 图

## Relational Database Keyword Search Based on E-R Diagram

Li Wei<sup>1</sup>, Gao Jin-Tao<sup>2</sup>, Gao Teng<sup>2</sup>

<sup>1</sup>(College of Computer Science and Technology, Shandong Jianzhu University, Jinan 250101, China)

<sup>2</sup>(Shandong Inzone Shopping Center Ltd. Jinan 250012, China)

**Abstract:** With the development of information retrieval technology and database theory, how to achieve information retrieval technology in relational databases to obtain useful information, and the normal user can also be performed in relation database query with keywords, as recent research hot spot. Relation Database query system based on keyword, with which the user don't need to know the database schema and structured query language, just inputting a few keywords. The article based on a prototype relational database-ER diagram, grouping entity according to different meaning. Following the keywords belonging to relation table, we can ensure the set of query entity, and dividing the range of query, to improve the efficiency of query. This paper presents a new algorithm to improve system precision. Finally, through experiments on the E-RKS system verifies verification.

**Key words:** relational database; keyword search; E-R diagram

随着信息检索技术的发展和人们查询需求的日渐提高, 查询行为已成为人们在线时十分重要的行为, 占到了全部行为的 21%, 其中对结构化数据的查询是主要的发展趋势<sup>[14]</sup>. 结构化数据存储的关系数据库的内部, 被称为 Deep Web, 搜索引擎不能直接获取这部分数据, 而这部分数据往往又是最为真实和准确的数据. 关系数据库内部结构化数据的数据量庞大, 并有固定的数据库模式, 而且需要结构化查询语言进行准确的查询, 这便限制了普通用户获取和操作此类数据,

所以对关系数据库中结构化数据的查询在最近十年一直是研究的热点.

基于关键字的关系数据库查询(Keyword Search Over Relational Databases, 简称 KSORD)系统, 不需要用户了解关系数据库的模式和结构化查询语言的语法, 仅需要几个关键字就可以返回查询结果, 满足了用户的查询需求.

KSORD 的实现方式可以分为基于模式图和基于数据图两种. 基于模式图的系统包括 DBXplore<sup>[1]</sup>, DISCOVER<sup>[3]</sup>,

<sup>①</sup> 收稿时间:2011-12-15;收到修改稿时间:2012-03-11

IR-Style<sup>[4]</sup>, Sqak<sup>[5]</sup>等; 基于数据图的系统包括BANKS<sup>[2]</sup>, BANKS-II<sup>[6]</sup>, ObjectRank<sup>[11]</sup>等. 基于模式图的方式是先找到包含关键字的关系表与它们间的连接, 也就是候选网络(Candidate Network, CN), 再把可生成的元组连接树作为结果返回给用户; 而基于数据图的方式是把所有的元组以及它们的连接生成图存入内存, 再利用多个关键字间最短路径算法找到合适的连接, 生成元组连接树. 在 DBXplorer, BANKS, DISCOVER 等系统中, 查询结果需要包含所有的关键字, 而 IR-Style 等系统不要求包含所有的关键字.

虽然现有的查询系统已经基本解决了基于关键字的关系数据库的查询问题, 但仍面临着以下三方面的问题:

① 现在关系数据库中数据量越来越庞大, 结构也越来越复杂, 现有查询系统的查询效率和查全率都还不足以很好的应用于实际.

② 现有的系统在对数据库建立图时, 都忽视了这些关系表的本质类型, 导致在每次查询时都有几乎遍历整个图. 其实这些关系表有些是实体类型, 有些是关系类型, 而用户需要的信息可能仅仅包含在其中的几个实体关系中, 在查询时把遍历范围控制在这几个实体关系中, 就会大大提高查询效率.

③ 结果返回机制不健全, 界面不友好一直是 KSORD 系统难以应用于实际的瓶颈. 返回结果是元组连接树, 没有很好的自然语言描述, 可读性不高.

针对以上的问题, 本文提出了一个基于 E-R 图的关系数据库关键字查询系统 E-RKS(E-R-based Keyword Search over relational database). E-R 图作为关系数据库结构的原型, 可以让我们真正了解关系数据库内部的关系. 通过使用 E-R 图, 我们可以把整个模式图分为几个实体关系的组合, 每次把查询范围限定在可能出现的小范围内, 便可以从根本上缩短查询时间, 提高查询效率. E-RKS 对 CN 生成算法进行改进, 避免了漏查情况的发生, 提高了结果的查全率. 在结果返回方面, 利用离线数据库实现 forms<sup>[10]</sup>返回机制, 进行对元组连接树向自然语言的转换, 以自然语言和元组信息两种信息返回给用户.

本文第 1 部分介绍出现的基本概念; 我们提出的 E-RKS 的系统结构会在第 2 部分进行介绍; 第 3 部分详细介绍 E-RKS 中的方法和算法; 实验在第 4 部分进行介绍; 第 5 部分是本文相关的工作; 第 6 部分是总结和未来工作.

## 1 基本概念

这一部分, 我们定义在描述 E-RKS 系统时会出现的概念.

定义 1 (模式图). 模式图用来描述关系数据库的内部结构, 可以表示为一个图  $G=(V, E)$  的形式,  $V$  是图中节点的集合, 一个节点  $v \in V$  表示数据库中的一个关系表;  $(v_1, v_2)$  代表图中的一条边,  $E$  是边的集合, 表示关系表间的连接关系, 也就是两个关系表间存在的主外键关系. 模式图  $G$  根据算法的需要, 可以表示为无向图, 也可以表示为有向图(其中正向是指主键所在关系表到外键所在关系表的连接方向). 图 1 所示为 TPC-H 数据库的模式图(有向图).

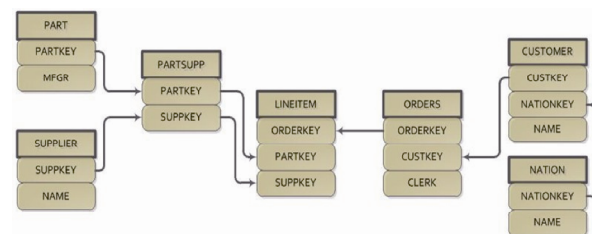


图 1 TPC-H 数据库模式图

定义 2 (关键字查询). 用户输入的一组关键字  $k_1, k_2 \dots k_n$ , 记作关键字集合  $Q=(k_1, k_2 \dots k_n)$ . 关键字查询便是用户向系统提交关键字集合  $Q$ , 并期望获得查询结果的过程.

E-RKS 系统会根据初始建立的全文索引, 确定  $Q$  中每个关键字对应的关系表以及属性, 并记录为关键字节点  $Vknrq(r$  指关系表,  $q$  指属性), 此类节点在查询时会被加入到模式图中.

定义 3 (候选网络 CN). 候选网络是指包含所有关键字的多个关系表(包含同一个关键字的同一个关系表在一条 CN 中只能出现一次), 通过主外键关系连接生成的关系树, 是模式图的一个子图. 该关系树中可能包括不含关键字的关系表, 用来提供其它关系表的过渡连接.

定义 4 (结果树). 结果树被表示为元组连接树的形式, 也就是一棵有根无序的树, 节点的信息中包含  $Q$  中的所有关键字, 其中叶子节点必须包含关键字.

元组连接树的长度为  $L$ (元组连接数, 最大定义为 6), 其大小体现了关键字之间的紧密程度. 元组连接树越长, 关键字间紧密度越低.

## ORDERS

ORDERKEY	CUSTKEY	CLERK
1000105	12312	John Smith
1000111	12312	Mike Miller

CUSTOMER

CUSTKEY	NAME	NATIONKEY
12312	Brad Lou	01

NATION

NATIONKEY	NAME
01	USA

LINEITEM

ORDERKEY	PARTKEY	SUPPKEY
1000105	1122	111222

PARTSUPP

PARTKEY	SUPPKEY
1122	111222

PART

PARTKEY	MFGR
1122	FST

图 2 简单的 TPC-H 数据库实例

CUSTKEY	NAME	NATIONKEY
12312	Brad Lou	01

ORDERKEY	CUSTKEY	CLERK
1000111	12312	Mike Miller

图 3 Q{brad, Smith}的其中一个结果树

例如, 在 TPC-H 数据库中, 给出关键字集合 Q{brad, Smith}, 得到其中一个结果(CUSTOMERNAME=Brad Lou $\infty$ ORDERSCLERK=MikeMiller), 结果树如图 3. 此结果树含义是顾客 Brad Lou 是店员 Mike Miller 为其提供服务.

定义 5(E-R 图实体集合分组). E-R 图为实体-关系图, 描述现实世界的概念模型, 真实的表现出了关系数据库的内部结构和关系. 图 4 就是 TPC-H 数据库

的 E-R 简图, 由图易得关系表 ORDERS、CUSTOMER、SUPPLIER、PART、NATION 为实体类型, 而 PARTSUPP、LINEITEM 为关系类型. 图中展现了两个具有明显不同意义的集合, 供应商与订单集合(SUPPLIER、PART、PARTSUPP、LINEITEM、ORDERS)和顾客与订单集合(ORDERS、CUSTOMER、NATION), 两个结合的交点为 ORDERS 实体. 我们利用 E-R 图重现了实体间的相互联系和紧密程度. 而且对于系统来说, 查询庞大的数据量并不比在一个特定的实体集合内查询更有意义. 当用户进行关键字查询时, 一般只会涉及到其中一个集合的信息, 毕竟顾客和供应商之间并不存在必然的联系.

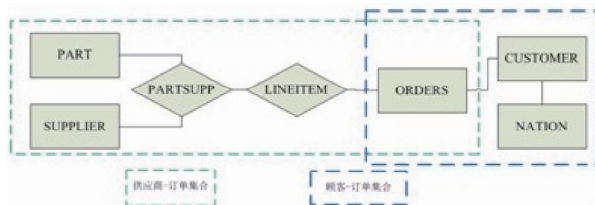


图 4 TPC-H 数据库 E-R 简图及实体集合分组

因此我们把查询的范围缩小到一个实体集合中, 会提高查询效率, 进而采用新的 CN 改进算法生成候选网络. 当关键字同时分散在多个集合时, E-RKS 系统也提供了另一种查询方法 CN 组合算法, 两种算法会在第 3 部分进行介绍.

E-R 图集合分组另一个优点在于其数据安全性. 当受限用户(只需了解消费订单信息或供货订单信息时)进行关系数据库查询时, 把查询范围限定在固定的实体集合内, 可以防止数据的泄露.

## 2 系统结构

E-RKS 系统分为三个模块: 预处理模块、查询模块和结果呈现模块. 系统结构图见图 5.

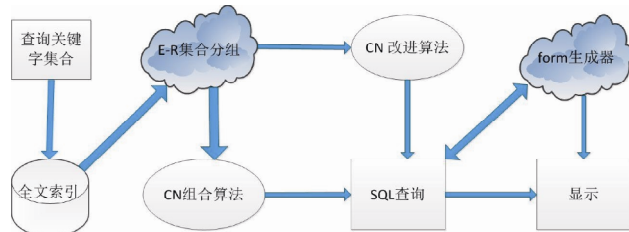


图 5 E-RKS 系统结构图

预处理模块首先对关系数据库进行预处理: 利用

RDBMS 对关系数据库建立全文索引,方便执行查询时,确定关键字所对应的关系表以及属性;然后利用关系数据库的结构原型 E-R 图对数据库关系进行实体集合分组.

查询模块是完成从在线处理用户输入关键字直到生成结果树的过程:首先利用全文索引确定 Q 中所有的关键字所属的关系表以及属性,记为节点  $V_{knrq}$ ,记入集合 S;然后根据 S 中包含不同关键字的节点的组合,确定此组合查询的实体集合,划定查询范围;在单独集合中查询使用 CN 改进算法,跨集合查询采用 CN 组合算法,全部查询完成获得 CNs;根据生成的 CNs,在关系数据库中生成 SQL,查询元组连接树,并把结果树提交给结果呈现模块.

结果呈现模块接收结果树,分析和生成可以描述其关键信息的自然语言语句 form,返回给用户.

### 3 E-RKS中的方法和算法

本部分主要介绍 E-RKS 系统中利用 E-R 图进行查询范围的划分, CN 改进算法和 CN 组合算法的描述,以及结果呈现机制.

#### 3.1 查询范围的划分

查询范围划分具体算法如下:

```

输入: 关键字组合集合 S, 顾客-订单集合 CO, 供
      应商-订单集合 SO
输出: S 中每个组合的范围划分
初始: C={}用于存储 CO 的划分, S={}存储 SO 的划
      分,
      CS={}存储多集合的划分
1. for S 组合的每一个组合 k
2.   if(k 中所有的关系表都属于 CO)
3.     则 K->C{};
4.   else if(k 中所有的关系表都属于 SO)
5.     则 K->S{};
6.   else K->CS{};
7. return C,S,CS

```

查询模块接收用户输入关键字后,会根据全文索引确定关键字所属关系表以及属性.例如查询关键字为 (Brad, Smith), 则会发现 Brad 在关系表 CUSTOMER(C) 中, 而 Smith 在 ORDERS(O) 和 SUPPLIER(S) 中都有出现, 则我们的查询就有了两种组合  $(C_{Brad}, O_{Smith})$  和  $(C_{Brad}, S_{Smith})$ .  $(C_{Brad}, O_{Smith})$  组合中关系表都存在于顾客-订单集合, 而  $(C_{Brad}, S_{Smith})$  则涉及到

顾客-订单与供应商-订单两个集合.

确定查询范围后,我们根据划分执行 CNs 生成算法.

#### 3.2 CN 改进算法与 CN 组合算法

##### 3.2.1 CN 改进算法

现有的 CN 算法在执行时可能出现漏查的现象,比如我们查询 (Brad, Smith, Miller) 时, 它的查询方法是找到 Brad 所在的 CUSTOMER 表, 利用主键 CUSTKEY 连接到 ORDERS 表的外键, 找到关键字 Smith  $(C_{Brad} \infty O_{Smith})$ , 但此时不能再反向连接到 CUSTOMER 表了, 因为不应该出现主键-外键-主键的这种连接  $(C_{Brad} \infty O_{Smith} \infty C_{Brad} \infty O_{Miller})$ , 定义 CN 时就避免了这种情况出现的. 故原有的 CN 算法可能会漏查这个结果.

本文提出的 CN 改进算法的主要思想是: 首先确定其中一个关键字所在位置作为根节点, 然后由根开始执行多路遍历来完成 CN 的生成.

其中根节点是由在数据库中出现次数最少<sup>[6]</sup>的一个关键字所在的位置确定, 这种根节点的语义更具有代表性, 能够反映查询结果的意义.

下面是 CN 改进算法:

```

输入: 模式子图(单个集合), 关键字集合
      Q(k1,k2...kn)
输出: CNs
初始: result{}
1.v->result{} //确定根节点 V,放入 result{}中
2.While Q.size()>1
3.   for R in result{} //result{}中元素为 R
4.     Rm 由节点 V 执行遍历模式图发现 kn;
5.     If(发现关键字)
6.       Rm=Rm+由 V 到 kn 遍历的关系表及路
          径;
7.       Rm->result{};
8.     else 由非根叶子节点执行遍历模式图发现
          kn;
9.       Rm=Rm+由非根叶子节点到 kn 遍历的关
          系表及路径;
10.    Q.size--;
11.return result{};

```

比如, 对于关键字集合 (Brad, Smith) 包括了顾客的名字和店员的名字, 其中顾客的名字一般各不相同, 而店员的名字可能在数据库中出现多次, 所以选择顾客的名字作为根节点进行查询, 并作为结果树的根节

点,更具有代表性.

### 3.2.2 CN 组合算法

CN 组合算法适用于关键字同时分散在多个集合中时执行. 关键字虽然散落在多个集合中,但它们通过集合间的交点还都存在着联系. 所以 CN 组合算法,首先从两集合的交点(TPC-H 数据库中指 ORDERAS 表)开始,向两个集合中的关键字分别执行最短路径算法. 然后合并重合的路径,得到结果树.

CN 组合算法如下:

```

输入: 模式图 G, 关键字组合集合 S, 两集合交点 V
输出: CNs
初始: result{}
1. for S 的每一个组合 k
2.   for 模式图 G 的两个子图
3.     R1(R2)=由 V 到关键字节点的关系表及路径;
4.     以 V 为根, R=R1+R2;
5.     R->result{};
6.return result{};
    
```

由以上两种算法生成的 CNs, 转化为 SQL 在 RDBMS 中执行生成结果树, 并提交给结果呈现模块.

### 3.3 结果呈现

分析后的结果树	form
ORDERS.CLARK.x--CUSTOMER.CUSTKEY--CUSTOMER.NAME.y	顾客y的订单由店员x来处理
ORDERS.CLARK.x--CUSTOMER.CUSTKEY--CUSTOMER.NAME.y --CUSTOMER.CUSTKEY --ORDERS.CLARK.z	顾客y的订单由店员x与店员z共同处理
CUSTOMER.NAME.x--CUSTOMER.CUSTKEY--ORDERS.CLARK.y --CUSTOMER.CUSTKEY --CUSTOMER.NAME.z	顾客x,z的订单都由y来处理
CUSTOMER.NAME.x--ORDERS.ORDERKEY--NATION.NATIONKEY --NATION.NAME.y	处理x顾客订单的店员是y国人
CUSTOMER.NAME.x--NATION.NATIONKEY--NATION.NAME.y	店员x是y国人

图 6 form 生成器

E-RKS 系统利用离线的 form 生成器把结果信息转化为自然语言呈现给用户, form 生成器中的内容是数据库管理员根据数据库自己定义的.

form 生成器在接收到查询模块提交的结果树后,会根据 Q 中的关键字来分析结果树的构成. 并用字符串匹配技术确定显示哪一条 form, form 生成器中对应关系如图 6.

例如查询关键字(Brad, Smith), (C<sub>Brad</sub>, O<sub>Smith</sub>)结果表示为“Brad Lou 的订单由 clerk John Smith 来处理”, (C<sub>Brad</sub>, S<sub>Smith</sub>)结果表示为“Brad Lou 的订单中显示商品来自于供应商 Smith 公司”. 利用 form 生成器我们可以很容易的获取原本难懂的元组连接信息.

## 4 实验

本部分对 E-RKS 系统的查询效率和查准率进行实验验证, 实验平台设置为 Dual-Core 2.70GHz 处理器、2G 内存、Windows 7 操作系统. 实验数据库为 TPC-H 数据库, 来自于网站 <http://www.tpc.org/tpch/>, 数据按照图 1 的数据库模式进行分解后装载在 SQLServer2005 数据库管理系统中. 本系统对 E-RKS 系统、DISCOVER 系统、IR-Style 系统进行查询效率和查准率两方面比较. E-RKS 系统代码全部由 java 实现.

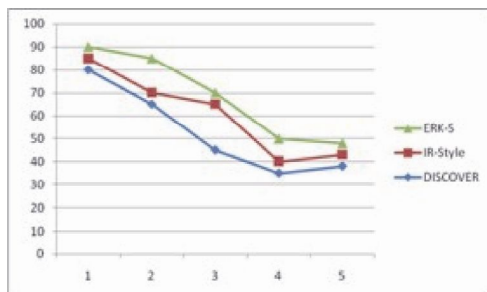
查询效率和查准率是衡量关键字查询系统查询效果的重要标准. 查询效率指单位时间内可以遍历的数据量的大小和获得查询结果的数量, 由查询结果的总量与数据集的数据量的比值获得; 查准率是指查询结果集中与关键字相关的查询结果所占的比重, 由查询结果集中与关键字相关的查询结果与查询结果集中查询结果的数量比值获得.

本实验的测试集由 30 个 SQL 查询组成, 并以在 RDBMS 中执行 SQL 的查询时间和查询结果为计算查询效率和查准率的基准. 实验过程中, 通过对三个系统输入相同的关键字, 实现与在 RDBMS 中执行 SQL 相同的查询目的, 记录查询时间和查询结果, 来计算 30 个 SQL 查询的平均查询效率和平均查准率.

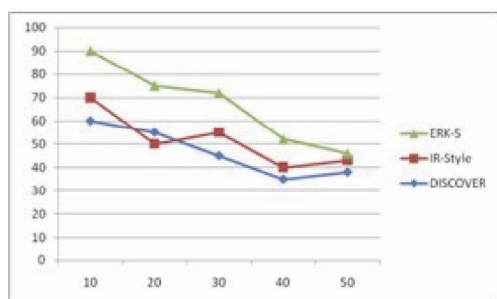
图 7(a)描述了 3 个系统在接收不同关键字个数时的平均查询效率. 由图看出在关键字个数较少时, 3 个系统的查询效率都很高, 但随着关键字个数的增多, 查询效率都有所下降, 但 E-RKS 系统对查询效率还是有一定的提高. 图 7(b)描述了 3 个系统在查询结果取样的前 50 个样本空间中的平均查准率比较. IR-Style 由于不要求结果



包含所有关键字,所以平均查准率较低,而 ERL-S 系统的平均查准率也较高.综上所述,ERL-S 系统提高了平均查询效率和平均查准率方面,切实改进了基于关键字的关系数据库查询系统的查询效果.



(a) 平均查询效率



(b) 平均查准率

图 7 实验结果比较

## 5 相关工作

基于关键字的关系数据库查询系统方便用户仅用简单的几个关键字就可以查询得到需要的数据库信息.其主要流程是把关系数据库转化为模式图,发现关键字所在关系表间可能存在的连接,并转化为 SQL 在 RDBMS 中进行查询返回结果.基于模式图的系统或算法主要有 DISCOVER,IR-Style, Sqak 等.

DISCOVER 系统首先采用 Oracle 的全文检索为每一个关系生成含不同关键字组合的元组集合,然后根据数据库模式图生成元组集合模式图,再从中找出包括全部关键字的子图,分析所有子图是否含有相同连接,提取作为中间结果替代连接,以减少连接次数.而后将处理后的子图转换为 SQL 查询,最后查询数据库得到查询结果,并根据元组连接数的多少来进行排序,有序的返回结果.

IR-style 系统是将信息检索领域已经很成熟的排序策略运用于数据库查询中,对 DISCOVER 系统进行

改进.该系统有以下功能:首先实现了“AND”和“OR”语义的同时处理,并引入了 IR 排序策略,使排序结果更令人满意,而且为提高查询效率,仅返回评分最高的 K 个结果,称为“Top-k”,最后返回的结果不再要求包含所有关键字.

除此之外研究者还是逐步完善基于关键字的关键字查询系统. Sqak 系统弥补了其他查询系统不支持聚集查询的不足. EKSO<sup>[13]</sup>, BLINKS<sup>[8]</sup>, EASE<sup>[9]</sup>等系统都采用了建立特殊索引的形式来提升查询效率. Tastier<sup>[7]</sup>系统可以在用户输入时就进行字符串匹配,不但给予用户输入提示,而且同时完成了关键字的查询. keyword++<sup>[12]</sup>等系统开始支持模糊查询.

## 6 总结和未来工作

E-RKS 系统通过对关系数据库进行还原得到 E-R 图,并进行实体集合分类,在关键字查询时只需在特定集合中进行遍历,缩小了查询范围,提高了查询效率. E-RKS 系统中的 CN 改进算法也防止了漏查情况的发生.结果呈现采用自然语言的形式,有更好的用户友好性.

E-RKS 系统在评分机制上还存在不足,还不能很好的平衡两种 CN 生成算法得到的结果,在未来的工作中会重点研究评分机制.

## 参考文献

- 1 Agrawal S, Chaudhuri S, Das G. DBXplorer: A System for Keyword-Based Search over Relational Databases. ICDE 2002:5-16.
- 2 Bhalotia G, Hulgeri A, Nakhe C, Chakrabarti S, Sudarshan S. Keyword Searching and Browsing in Databases using BANKS. ICDE 2002: 431-440.
- 3 Hristidis V, Papakonstantinou Y. DISCOVER: Key-word Search in Relational Databases. VLDB 2002:670-681.
- 4 Hristidis V, Gravano L, Papakonstantinou Y. Efficient IR-Style Keyword Search over Relational Databases. VLDB. 2003: 850-861.
- 5 Tata S, Lohman GM. SQAK:doing more with keywords. SIGMOD. 2008: 889-902.
- 6 Kacholia V, Pandit S, Chakrabarti S, Sudarshan S, Desai R, Karambelkar H. Bidirectional Expansion For Keyword Search on Graph Databases. VLDB. 2005: 505-516.
- 7 Li GL, Ji SY, Li C, Feng JH. Efficient type-ahead search on

(下转第 140 页)

## ① 配置同步周期和同步服务器

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\W32Time\TimeProviders\NtpClient 选项中的 SpecialPollInterval 字段配置同步周期, SpecialPollTimeRemaining 字段指定同步服务器。

## ② 修改默认更新服务器

KEY\_LOCAL\_MACHINE\SOFTWARE\MICROSOFT\WINDOWS\CURRENTVERSION\DATETIME\SERVICES\default 字段指定默认时间同步服务器。

③ 重启 Win32Time 服务: net stop w32Time && net start w32Time.

## 2.3.4 PLC 系统

各站场 PLC 系统利用 PLC 机架上的以太网模块与站场的操作员站实现定时<sup>[8]</sup>。我们选用 RSLogix5000 系列 PLC, 使用其 RSLgoix5000 Clock Synchronization Tool 时间同步软件, 安装在站场的操作员站上, 利用通用工业控制协议(CIP 协议), 对指定 PLC 的 CPU 时间进行修改, 同步周期可以自由设定。

对于管网 SCADA 系统, 采用软硬件结合的方法实现了多级服务器分级定时, 可以有效避免某一时间服务器故障, 导致整个系统的 NTP 同步体系瘫痪, 一般而言, 长时间的高层级的时间服务器的故障, 导致网络设备出现时间误差超过 1000 秒, NTP 协议将不能实现自动定时, 在实际中, 需要定期对 NTP 进行测试。

## 3 结语

本文介绍了基于软硬件同步相结合的时间同步方法, 采用了成熟的产品和技术, 已经成功应用于管道 SCADA 系统的生产运行中。利用该方案后, 可以有效的实现各个网络设备的工作时间同步, 确保系统运行的稳定可靠, 可以向相应的企业网络进行推广应用。

## 参考文献

- 1 沈燕芬.用于网络时间同步的 NTP 协议.现代计算机,2004,4:54-56.
- 2 谢华.基于 GPS 信号和 NTP 协议的本地时间同步网络.工业控制计算机,2010,23(6):75-76.
- 3 艾艳.GPS 授时服务系统在气象业务中的应用.气象与环境科学,2008,31(4):85-87.
- 4 赵英,徐金平,韩少延.网络时钟同步系统的设计.计算机应用,2004,24(增刊):12-14.
- 5 贾平平,艾艳,杨立志,卫权岗,任宏,刘静.基于 NTP 的网络定时系统设计与实现.气象与环境科学,2010,33(4):89-91.
- 6 方振伟,王朝阳,孙开云.中小企业数据中心实现 NTP 时间同步实例.微电脑世界,2011,11:78-81.
- 7 杨先杰.NTP 协议的研究与应用.电力信息化,2011,6:28-32.
- 8 张宁宁,姜波.ControlLogix 系统在实现系统 CPU 定时中的应用.工业控制计算机,2008,21(3):93-94.

(上接第 161 页)

- relational data: a TASTIER approach. SIGMOD 2009: 695-706.
- 8 He H, Wang HX, Yang J, Yu PS. BLINKS: ranked keyword searches on graphs. SIGMOD 2007: 305-316.
  - 9 Li GL, Ooi BC, Feng JH, Wang JY, Zhou LZ. EASE: an effective 3-in-1 keyword search method for unstructured, semi-structured and structured data. SIGMOD 2008: 903-914.
  - 10 Chu E, Baid A, Chai XY, Doan AH, Naughton JF. Combining keyword search and forms for ad hoc querying of databases. SIGMOD. 2009: 349-360.

- 11 Balmin A, Hristidis V, Papakonstantinou Y. Object Rank: Authority-Based Keyword Search in Databases. VLDB 2004: 564-575.
- 12 Xin D, He YY, Ganti V. Keyword ++: A Framework to Improve Keyword Search Over Entity Databases. PVLDB, 2010,3(1):711-722.
- 13 Su Q, Widom J. Indexing Relational Database Content Offline for Efficient Keyword-Based Search. IDEAS 2005: 297-306.
- 14 Kumar R, Tomkins A. A characterization of online browsing behavior. WWW 2010: 561-570.