

文章编号: 2096-1472(2016)-03-01-04

一种面向效率 and 安全性同时提高的数据库优化方法

刘东明

(哈尔滨工业大学(威海)计算机科学与技术学院, 山东 威海 264209)

摘要:常规的数据库应用受制于字段结构的限制,虽然简化了处理流程,但处理效率比较低,往往形成应用的瓶颈。使用联合字段对数据库结构进行优化,使原有的重复数据得以整合,结合SQL语言与单一类型存储,可以有效提升数据的检索效率,快速产生初级可选集。对可选集的操作可以使用原有的数据库系统,也可以使用定制的数据库,定制数据库不但可以摆脱数据库系统对开发过程的限制,而且在数据的处理与加解密方面更加主动灵活,是小型数据库应用开发的一个趋势。

关键词:数据库结构;联合字段;优化;检索效率;定制数据库

中图分类号: TP392 **文献标识码:** A

A Plan of Enhance the Retrieval Efficiency by Optimization Data Organization

LIU Dongming

(The Computer Department of Harbin Institute of Technology in Weihai, Weihai 264209, China)

Abstract:Conventional database application is restricted by the field structure,although simplified the processing flow,but the processing efficiency is low,often the bottleneck in the application.Using united fields to optimize the database structure.The original repeating data can be integrated,with the combination of SQL language and the single type of storage,can improve the efficiency of data retrieval,fast generation primary optional set.On the optional set operation can use the original database system,can also use the custom database.Customize database can not only get rid of the limitation of database system development process,but also more initiative and flexibility in the data processing and encryption and decryption,is a development trend in small database application.

Keywords:database structure;united fields;optimize retrieval efficiency;customize database

1 引言(Introduction)

近年来,随着计算机硬件设备的不断升级,软件处理能力显著提高。然而,在数据库处理方面,却始终存在着一个瓶颈,那就是数据库的字段结构一直制约着数据的处理速度。目前数据库的主要存储方式是硬盘设备,而硬盘设备本身存在着处理速度的极限。在这种条件下,如何提高数据的检索速度就成为一个迫切要解决的问题,尤其是在数据库容量越来越大的情况下。

数据检索的核心问题在于,如何在海量数据(硬盘存储)中检索出符合条件的可选集,而不在于对于可选集的处理(内存)上,因为内存的访问速度对于数据检索来说是足够的。一旦从数据库上获得可选集,后继的工作就可以转移到内存中工作,处理器的性能就可以得到充分发挥。因此,在提高数据检索的速度方面,应该在优化数据库结构,快速获得可选集上面下工夫,其中包括:减少数据库字段与记录的数量、简化数据类型、减少数据冗余等方面^[1]。

2 数据库结构的优化(Optimization of database structure)

常规的数据库字段是根据数据的内容属性来定义的,用

户对数据的操作界面与数据库结构保持一致,以简单的人员信息管理为例,数据库结构如表1所示。

表1 简单的人员信息数据库结构

Tab.1 Database structure of staff

编号	姓名	性别
001	王勇	男
002	李小明	女
...		

这是一种通用的途径,也是数据库厂商建议使用的方法,因为这样可以直接使用数据库访问的预定义功能,包括用户操作界面,同时简化数据库应用程序的开发工作,把程序开发人员和操作人员与核心数据库隔离开来,充分保证数据的安全性^[2]。但是,这种使用方法也制约了数据库开发人员的视野,尤其在大型数据库的应用开发过程中,在数据检索的效率方面显得无所作为,或者在具体应用出现瓶颈时推卸责任。

因此,在数据库开发的规划方面,不一定严格遵循数据库厂商建议的方式,可以根据不同的应用场合和数据类型,建立灵活的数据结构,从而提高数据检索的效率^[3,4]。

这里给出的一种方案,是在数据库结构方面进行的一个大胆的尝试,对于数据类型单一、数据容量较大的应用场合较为适用。它不按数据的内容属性来定义具体的字段,而是采取“联合字段”的结构,将原来多个单一的字段组合成一个联合字段,从而简化了数据库的结构,大大提高了可选集的产生过程可数据检索的效率。

表1中的人员信息数据库,变换为组成联合字段以后,结构简化如表2所示。

表2 变换联合字段的数据库结构

Tab.2 United fields database structure

人员信息
-001王勇男-002李小萌女-...

从表2可以看出,原来数据库中的一张表,现在被简化成了一条记录。通过这种变换,甚至可以把整个数据库都整合在一个字段中,将针对数据库的操作转换为针对少量字符串的操作,大大简化数据的操作过程,提高数据的检索效率。

在这种方案中,当可选集产生以后,不再使用数据库系统提供的缺省操作界面,而由数据库应用程序开发人员编制特定的操作界面,当然,从外观上,可以借鉴操作人员已经习惯的系统界面,甚至与系统界面保持一致,但下面的代码完全由开发人员自己来完成。这样的开发工作并不复杂,不管在哪种平台上,都不会给开发人员增加太多的额外工作量。而且,程序使用效率的大幅提高,也会很快将额外的开发工作量弥补掉。

联合字段的数据工作过程如图1所示。

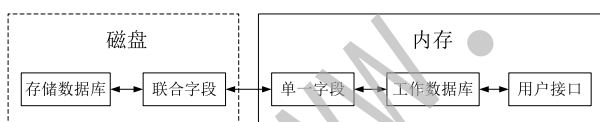


图1 联合字段的数据工作过程

Fig.1 Data working procedure for united fields

虽然在界面上可以与数据库系统保持一致,但数据库接口与数据库结构对用户来说是完全不透明的,从这个角度来看,不但没有降低数据的安全性,而且在数据转换与存储的过程中,还可以采取灵活的手段进行加解密,提高原有数据的安全性。由于变换后的数据对数据厂商也是不透明的,因此,就实现了数据的双向安全性,这正是目前几乎所有大型数据库应用系统一直面临的困惑。

联合字段实现双向安全性,如图2所示。

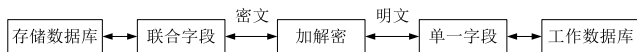


图2 联合字段实现双向安全性

Fig.2 United fields to achieve two-way security

联合字段的作用,是简化可选集的产生过程,提高可选集的检索效率,因为针对硬盘存储的数据库检索,也是以字段为最小单位进行的,字段数量的缩减,以及字段类型的单一化,对于可选集检索效率的贡献是可想而知的。

3 数据的处理过程(Data processing procedure)

3.1 数据单元的概念

为了阐述方便,首先给出“数据单元”的概念,把按规律排列的格式整齐的重复性的一组数据称为一个数据单元,在常规的数据库中,数据单元也许是一条记录,也许是一条记录中的几个字段^[5]。例如在上面的人员信息数据库中,每个人员的信息就可以作为一个数据单元,如:“001王勇男”,姓名在中间,左侧是编号(占三个字符),右侧是性别(占两个字符),格式整齐,排列有规律。在这样的一个数据单元中,编号数据可以从单元数据的左侧选取,性别数据可以从单元的右侧选取,中间剩下的就是姓名的数据,这样的选取方法,不要求位于单元中间的姓名数据为特定的长度。

利用数据单元,可以对原有的字段进行初步的整合,减少字段的数量,简化字段的类型,精简数据库的原有结构。这样变换后,并不影响对原有数据库的常规操作,因为用户界面是根据功能而重新开发的,不依赖于数据库系统,它所使用的是内存中的数据,已经把磁盘中的存储数据库恢复为原有结构的工作数据库^[6]。工作数据库根据操作需要,只选取全部数据的一个子集,所以,对于处理速度与实时性的影响有限,这与系统数据库的工作原理是一样的,见图1。只是在得到可选集的条件上,要进行相应的变换,由于数据单元并不复杂,而且是大量的重复性数据,所以在原理上可以保证可选集的确定性。

如果数据单元的变换已经能够满足应用的需要,优化过程就可以就此为止,有时候,为了简化开发工作,甚至可以将工作数据库与存储数据库保持一致,这样,应用效率的提高,只体现在可选集的产生过程中,尽管如此,变换前后的效率差异还是显而易见的。

3.2 数据整合

如果要进一步提升系统的功能,可以在数据单元的基础上,在两个层面上继续对数据进行整合^[7]。

第一个层面,对数据单元进行整合。把多个数据单元连接在一起,进行联合操作。实现的方法是多种多样的,比如,可以使用特定的分隔连接符,把单元数据连接起来,上面的人员信息数据库,整合以后简化成一条记录:

001王勇男-002李小萌女-003张虹男-...

上例中分隔符的选择不能与数据库中的任何数据重复,

当数据库的性质与内容确定后，找到合适的分隔符并不困难，它的作用只是单纯地将数据单元分隔开来，并且在数据处理以双向过程中作为定位符，实现工作数据库与存储数据库之间的数据传递。

第二个层面，是对字段进行联合。这可以打破数据库系统对字段容量的限制。数据单元整合后得到的数据，具有单一的字符串类型，跨字段存储只涉及到简单的分割问题，这是稍有编程经验的开发人员都可以完成的工作。在数据处理的过程中，直接对多个字段进行联合操作，处理过程并不复杂，由自主开发的用户接口程序完成这一工作，应用人员不必关心其中的细节。

数据库系统在进行数据处理时，往往以字符类型为主，如果数据库中存在多种数据类型，在进行数据处理时也会在内部先转换为字符类型，然后再进行相应的操作。对数据库结构进行优化后，联合字段类型可以全部设置为字符串类型，这样就简化了内部的类型转换，提高了处理效率，检索速度也是最快的。

要注意的是，最好不要让整合后的联合数据超过操作系统的平台所允许的长度，否则在开发时就要面临来自操作系统的种种限制，会影响到用户程序开发的效率。

3.3 整合数据的处理

对整合数据进行处理，主要是在工作数据库与存储数据库之间建立起联系的通道，数据检索时，根据指定的条件，首先从存储数据库产生可选集，并以此为基础建立工作数据库，其后的大部分操作都是面向工作数据库的，在内存中完成，只有在操作结束后，或者数据出现变动的情况下，才与存储数据库进行数据交换^[8]。

例如，在人员信息数据库中，按姓名查找某个人，直接输入查询条件，将优化后的联合字段作为操作对象即可。如果有多个查询条件，比如姓名与性别，可以先对条件进行组合，把“姓名+性别”特征串作为查询条件，然后在存储数据库中进行检索。

由于存储数据库中的字段已经被优化，字段数量减少、存储类型单一，因此，从存储数据库中检索出可选集的效率得到大幅度提高。

3.4 注意事项

(1)使用SQL查询语言

在可选集的产生过程中，应该充分利用数据系统的技术支持，比如SQL查询技术及数据库索引等技术。即使数据库结构保持不变，使用SQL查询语言，也可以成倍提高检索的效率。在具体的检索过程中，数据库系统会自动建立索引文

件，进一步提升数据检索的速度。经初步测算，使用SQL查询语言比不使用在速度上可以提高几十倍。

(2)用户界面与工作数据库

由于数据库系统对用户来说是不透明的，所以，应该在数据库应用层面上建立特定的用户接口程序，上面已经提到，这部分工作是传统的数据库开发工作之外的额外工作，主要完成可选集的建立工作，与优化后的数据库结构相匹配。此类的开发工作并不复杂，主要内容是进行大量的字符串变换。用户接口程序是双向工作的，在工作数据库与存储数据库之间传递用户的数据。应用系统对数据操作完成以后，用户界面负责临时数据库中的内容交换到存储数据库中，这个过程只在数据存在变换时进行。在用户操作的过程中，使用的数据库是临时的工作数据库，可以建立在原有数据库系统的基础上，使用临时数据库或者数据快照，把整合的数据重新恢复成内容属性的字段，这是与存储数据库无关的过程，只不过针对的是内存中的临时数据库。因此，如果只是使用现有的数据，而不进行数据的改动，这部分操作界面的实现相对简单^[9]。

4 使用定制数据库(Using custom database)

如果数据规模不大，但对安全性要求较高，可以彻底摆脱数据库系统的限制，采取用户定制的数据库，从底层上直接对数据进行操作，同时便于数据的加解密。目前这类系统在国内的应用不多，也没有引起大多数应用人员的重视。近年来，随着网络安全形势的日益恶化，对于涉及国家重大利益的数据操作，安全性已经成为系统开发的首要目标。因此，使用定制数据库与密文存储的技术，将是未来一段时间内数据库应用开发的重要方向。

定制数据库的工作过程如图3所示。

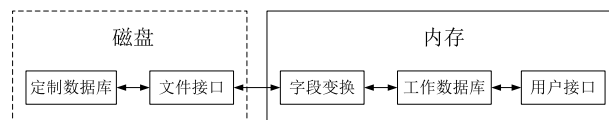


图3 用户定制数据库的工作过程

Fig.3 Data working procedure for custom database

4.1 存储形式

定制数据库，可以用磁盘文件的形式直接存储在物理介质上。如果数据量不大，可以将多张数据表格放在一个文件中，以简化存储结构。如果数据量较大，可以将数据表格单独存储，以提高数据处理的速度。

文件的格式可以采用BINARY或者ASCII编码的形式，使用开发平台下的文件存取指令进行操作。在文件的头部或尾部，添加指定的数据表及数据块的索引信息，用以标明文

件的数据属性。

在数据的组织上,仍然沿用联合字段的思想,只不过不再借用数据库系统的字段结构,直接与磁盘文件进行数据交换。这样就省去了数据的间接操作过程,在底层上控制数据的存储,增加了数据存储与处理的灵活性,提高了系统的工作效率。

在此类应用中,要求开发人员对数据库系统的结构有大致地了解,除了常规的文件操作外,还应考虑大量的容灾与安分方面的性能。由于对磁盘文件的操作只包括存储与产生可选集两个方面,类似的开发过程肯定比专业的数据库系统简化,否则就失去了定制开发的意义。

4.2 数据处理的过程

由于底层数据不再依赖数据库系统的支持,在可选集产生之后,工作数据库也没有必要再使用数据库系统提供的处理功能,而转由常规的数据结构来实现,比如使用动态数组或容器等结构。

这样,在用户界面的开发方面,就具有了更大的空间和灵活性,打包后的应用系统也更加精简,便于移植与发行,拓宽了应用系统的应用领域,特别是对于一些安全性要求较高的场合较为适用。

在具体数据的处理过程中,仍然可以使用前面介绍的方法,只是在用户界面的修饰方面,要下一定的功夫,达到与专业数据库系统同样的水平,这方面的开发工作不影响具体的数据应用,只是照顾用户的原有操作习惯。

4.3 加解密过程

定制的本地数据库,只对存储环节进行加解密,正常的使用过程,只有数据变动时才需要与存储数据库进行数据交换,大部分操作都是针对内存中的工作(临时)数据库进行的。此类数据加解密过程如图4所示。

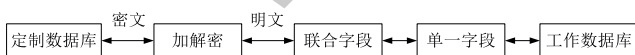


图4 定制本地数据库的加解密过程

Fig.4 Encryption and decryption process of customized local database

定制的网络数据库,可以通过直接调用底层的网络传输协议进行访问,数据加解密的过程在全部在客户端实现,经过网络传输的数据都是加密后的密文,这样就有效确保了数据传输过程的安全性,也不必考虑众多的网络安全设备(如网络防火墙、网络安全网关等)的影响。

定制的网络数据库的加解密过程如图5所示。

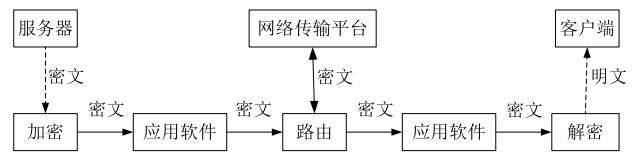


图5 定制网络数据库的加解密过程

Fig.5 Encryption and decryption process of customized network database

4.4 定制数据库的开发环境

定制的数据数据库不依赖于商业化的数据库系统,直接对磁盘文件进行读写。因此,在开发环境的选择方面更加自由,包括普通的桌面系统、网络化的C/S、B/S系统、移动(手机、平板)平台等,都可以完成有效的开发工作。

而且,当基本的应用流程固定以后,可以在不同的平台之间实现快速移植。一个平台上开发完成的代码,只需改动少量的外围代码,就可以应用到另一个平台上。这样,就彻底摆脱了平台软件与数据库系统对用户程序的开发限制,也减少了庞大的软件与系统资源开销,使开发工作变得更加简单灵活。

IT产业发展到现阶段,硬件的性能得到了很大的提升,原来依靠专业数据库来支持的大型数据库应用系统,现在如果转而使用定制的数据数据库,可以将其部署在普通的硬件环境之上,几百万上千万条的数据容量,可以轻而易举地在新平台上流畅运行。

在移动终端平台上,使用定制的数据数据库,能够大大减少数据库系统占用的开销,转而支持新的应用项目,使有效的硬件资源得到充分合理的运用。

5 应用实例 (Application examples)

以上技术最初应用于《列车时刻表查询软件V4.0》(软著登字第0002941号)中,经实际检测,使用联合字段对数据库结构进行优化以后,数据检索的效率比优化之前提高了70多倍,较好地解决了数据检索速度的瓶颈。下面简要介绍应用的过程。

(1)问题分析

交通信息的数据具有统一的规律,就是除了少数的数据以外,大部分数据都是重复性的信息,如到达某个站点的进出站时间与站名等信息。这种数据的特点,比较适合采用联合字段的优化方法。

如果按着常规的定义方法,每个班次的数据定义成一张独立的表格,如果实现区域联网,数据量将是惊人的。当对联网后的数据库进行模糊检索时,检索效率低下的问题表现得十分明显。因此,采用技术手段对数据库结构进行优化,是解决问题的一个途径。正是基于这样的思路,尝试将联合

字段技术应用在大范围交通信息数据库系统的开发过程中,将每个班次的交通信息简化成单条记录,所有区域联网的班次都纳入一张数据表格中,大大简化了数据存储与处理的效率,较好地解决了同类系统开发过程中长期存在的症结。

(2)数据单元

将中途站点的数据制作成数据单元,形式为:进站时间+站点名称+出站时间,进出站时间的字符长度直接固定为5,如12:48,因此,数据单元的首尾长度都是确定的,只有中间的站点名称是不确定的。始发站点与终到站点的数据不用特殊处理,直接按上述规则放入数据中即可。

(3)数据编辑

数据单元之间使用常规数据中不会出现的特殊字符进行连接,如“-”字符,一个班次的数据按着字符形式存储在一条记录中。数据处理时,通过检索条件在存储库中产生初级可选集,然后对可选集进行具体的操作。

可选集的产生,可以使用数据库系统提供的功能,组合SQL查询条件。在可选集中,根据连接字符将记录中的数据单元分离出来。每个数据单元,先分离出首尾的时间数据,中间余下的就是站点名称数据,并依此建立临时工作数据库,与用户接口程序进行交互。数据变动后,将数据进行逆向处理,最终完成存储数据库的更新过程。

(4)实例:区间模糊检索

这是最常规的应用,只设定两个站点的名称,即可以检索出区间内的所有班次。检索条件可以组合为:记录中“前面包含‘出发站点名称’并且后面包含‘到达站点名称’”的可选集”。

(5)实例:生成站点的班次时刻表

时刻表是动态生成与显示的,可以直接上物理的大屏系统,为所有站点的出行人员提供信息服务。检索条件可以组合为:记录中“包含‘站点名称’”的可选集。在可选集中,根据进出站时间的特点,可以将班次的属性区分开来,如始发、途经与终到等班次。

(上接第30页)

必须考虑的问题,网络覆盖控制作为WSN实施过程中的一个基本问题,反映了网络所能提供的“感知”服务质量^[5]。本文根据近年来的覆盖控制研究成果,对WSN的覆盖问题进行了归纳与总结,提出了研究WSN覆盖问题时所需关注的各个方面,并详细介绍了无线传感器网络算法。WSN的覆盖控制依然存在许多亟待解决的问题。例如,现有的WSN覆盖控制策略主要集中在解决简单传感模型下的覆盖控制,对于节点感知存在方向差异的情况的研究,仍然是一个比较复杂的难题。

参考文献(References)

- [1] 任彦,张思东,张宏科.无线传感器网络中覆盖控制理论与算法[J].软件学报,2006(3):422-433.
- [2] 曹峰,刘丽萍,王智.能量有效的无线移动传感器网络部署[J].

6 结论(Conclusion)

通过对数据库结构的优化与定制,使数据类型单一化,不但简化了存储形式,而且可以快速实现数据的加解密过程,加解密运算全部在客户端实现,充分发挥了本地处理器的处理能力,不影响网络数据的访问速度。

可见,数据结构优化后,简化了数据结构,有效提高了数据的处理效率与安全性,节省了存储空间,方便了数据的跨平台移植,是小型数据库应用系统开发的一种新思路,特别适合于以大量重复数据为主体的数据库应用场合。

参考文献(References)

- [1] 李宏伟.地名本体数据库存储模式及应用研究[J].计算机应用与软件,2012,29(4):35-38;74.
- [2] 陈正举.基于HIBERNATE的数据库访问优化[J].计算机应用与软件,2012,29(7):144-149.
- [3] 林洋.分布式数据库中空间拓扑连接查询优化处理方法研究[J].计算机应用与软件,2013,30(11):247-250;282.
- [4] 房俊华.DB-Tree:一种高性能的闪存数据库索引结构[J].计算机应用与软件,2013,30(11):243-246.
- [5] 陈芬.改进量子粒子群算法优化神经网络的数据库重复记录检测[J].计算机应用与软件,2014,31(3):20-21;115.
- [6] 林桂亚.基于粒子群算法的数据库查询优化[J].计算机应用研究,2012,29(3):947-949.
- [7] 王兵.数据库应用系统逻辑结构设计初探[J].现代计算机,2012,(5):14-17.
- [8] 赵荣.分布式数据库查询优化方法[J].科技视界,2013(5):120-121.
- [9] 曹永峰.一种新的查询优化方法[J].计算机与现代化,2005(7):64-69.

作者简介:

刘东明(1966-),男,硕士,副教授.研究领域:数据库,数字媒体技术.

信息与控制,2006(2):147-153.

- [3] 王燕莉,安世全.无线传感器网络的覆盖问题研究[J].传感技术学报,2005,18(2):307-312.
- [4] 任彦,张思东,张宏科.无线传感器网络三维空间最佳覆盖路由协议[J].电子学报,2006(2):306-311.

作者简介:

张永棠(1981-),男,硕士,副教授,系统分析师.研究领域:光通信与传感器网络应用.

罗先录(1973-),男,硕士,副教授,系统分析师.研究领域:软件与网络安全.

周富肯(1982-),男,硕士,讲师.研究领域:网络安全.