



(12) 发明专利申请

(10) 申请公布号 CN 117425886 A

(43) 申请公布日 2024. 01. 19

(21) 申请号 202280039959.9

(22) 申请日 2022.04.22

(30) 优先权数据

21170144.6 2021.04.23 EP

(85) PCT国际申请进入国家阶段日

2023.12.04

(86) PCT国际申请的申请数据

PCT/EP2022/060747 2022.04.22

(87) PCT国际申请的公布数据

WO2022/223807 DE 2022.10.27

(71) 申请人 皮质创新有限公司

地址 德国伊森哈根

(72) 发明人 彼得·帕尔姆

(74) 专利代理机构 北京高沃律师事务所 11569

专利代理师 万慧华

(51) Int.Cl.

G06F 16/21 (2006.01)

权利要求书7页 说明书36页 附图11页

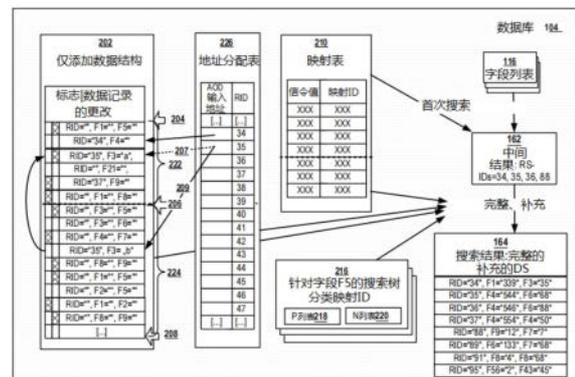
(54) 发明名称

具有仅添加数据结构的基于列表的数据搜索

(57) 摘要

本发明涉及一种用于在数据库(104)内执行数据库查询的计算机实现方法,该数据库(104)包含作为字段特定数据值列表(116)物理地存储的逻辑记录,该方法包括:-接收指令,以便更改数据值;-在不更改字段特定数据值列表(116)的情况下,将指令存储(604)在仅添加数据结构(202)中,仅添加数据结构中的每个条目在本文中被称为AOD条目并且至少包含根据更改指令之一要更改的记录之一的那些字段标识符数据值对;-对于以下数据记录中的每一个:即关于数据库接收到指令,在地址分配表(226)中存储(606)对所述记录的更改进行指定的被存储AOD条目中的最近AOD条目的地址(206、208),该记录与所述记录的记录ID相关联;以及-执行(610)数据库查询,数据库查询包括:i.搜索(612)字段特定数据值列表,以便识别记录(214)的ID;ii.评估(614)地址分配表,以便识别分配给在(i)中识别出的记

录ID之一的AOD条目的地址,iii.访问(616)识别出的AOD条目的地址;以及iv.使用(618)包含在所述识别出的AOD条目中的更改信息,以添加字段标识符数据值对并将其输出给在步骤i)中识别出的记录ID。



1. 一种用于对数据库(104)执行数据库查询的计算机实现方法,其中,所述数据库在被称为“最近合并时间点”的第一时间点处包含多个逻辑数据记录,其中,每个数据记录包含数据记录ID和一个或多个字段标识符-数据值对,其中,所述数据记录以字段特定数据值列表(116、402、404、406)的形式物理地存储,其中,所述方法包括:在所述最近合并时间点之后,

-接收(602)更改所述数据记录中若干数据记录的字段的数据值的指令;

-在不更改所述字段特定数据值列表(116)的情况下,将所述指令存储(604)在仅添加数据结构(202)中,其中,所述仅添加数据结构中的每个条目在此被称为AOD条目并且至少包含根据更改指令之一要更改的数据记录之一的字段标识符-数据值对中的那些条目;

-对于以下数据记录中的每一个:即关于该数据记录在所述最近合并时间点之后所述数据库接收到更改数据值的一个或多个指令,在地址分配表(226)中存储(606)对所述数据记录的更改进行指定的被存储AOD条目中的最近AOD条目的地址(206、208),所述数据记录链接到所述数据记录的数据记录ID,其中,所述地址分配表中的链接被自动更新;以及

-执行(610)数据库查询,其中,所述数据库查询包括:

i. 基于与一个或多个字段特定搜索值的匹配,搜索(612)所述字段特定数据值列表以识别内容要被全部或部分返回的数据记录(214)的ID;

ii. 访问(614)所述地址分配表以识别向在i)中识别出的所述数据记录ID之一分配的AOD条目的地址,

iii. 访问(616)识别出的AOD条目的地址;以及

iv. 使用(618)包含在这些识别出的AOD条目中的更改细节,以将字段标识符-数据值对添加到在步骤i)中确定的所述数据记录ID,并输出后者。

2. 根据权利要求1所述的计算机实现方法,

-其中,所述地址分配表中恰好一个条目对应于所述逻辑数据记录中的每一个;和/或

-其中,所述方法还包括由所述DMS系统协同管理所述逻辑数据记录和所述地址分配表中的条目,使得所述逻辑数据记录总是被生成并且与所述地址分配表同步,使得每个逻辑数据记录的ID显式地或隐式地指定所述地址分配表中该条目的存储器地址;在所述地址分配表中该逻辑数据记录的ID被分配具有对该数据记录的最近更改的所述AOD条目的地址,并且其中,特别是在步骤ii)中,在所述数据记录ID中指定的存储器地址用于直接访问所述地址分配表中被唯一地分配给在i)中确定的所述数据记录的那些条目。

3. 根据前述权利要求中任一项所述的计算机实现方法,还包括:

-在被称为“新合并时间点”的第二时间点处,通过关于对所述字段特定数据值列表执行数据库查询而独立地和/或并行地合并所述字段特定数据值列表,来合并自所述最近合并时间点以来指示的更改;或

-在被称为“新合并时间点”的第二时间点处,通过关于对所述字段特定数据值列表执行数据库查询而独立地和/或并行地生成所述字段特定数据值列表的合并副本(228),来合并自所述最近合并时间点以来指示的更改。

4. 根据权利要求3所述的计算机实现方法,还包括:

-对于所述字段特定数据值列表中的每一个的每个数据值:在被称为否定列表(220)的列表和数据记录ID中,存储由于在所述第一合并时间点和所述第二合并时间点之间接收到

的更改指令而不再被分配关于该字段的该数据值的所有逻辑数据记录的数据记录ID;

-对于所述字段特定数据值列表中的每一个的每个数据值以及对于所述字段中的每一个的新数据值:在被称为肯定列表(218)的列表和数据记录ID中,存储由于在所述第一合并时间点和所述第二合并时间点之间接收到的更改指令而要分配关于该字段的该数据值的所有逻辑数据记录的数据记录ID;

-其中,执行所述合并包括根据所述肯定列表和所述否定列表的内容更新所述字段特定数据值列表或所述字段特定数据值列表的副本的数据值和数据记录ID;以及

-其中,在合并之后清空所述肯定列表和所述否定列表。

5. 根据权利要求3或4所述的计算机实现方法,

-其中,所述方法包括关于所述合并并行地且独立地、不中断地对所述字段特定数据值列表执行进一步的数据查询;和/或

-其中,所述字段特定数据值列表在所述合并的执行期间也不锁定于包含在其中的数据值。

6. 根据权利要求3-5中任一项所述的计算机实现方法,其中,对所述字段特定数据值列表中的每一个执行所述合并包括:

-分析数据记录ID的字段特定和数据值特定的肯定列表(218)和否定列表(220),以确定与字段的唯一数据值的数量相关的更改以及与在所述第一合并时间点和所述第二合并时间点之间链接到该字段的数据值的数据记录ID的最高数量相关的更改;

-根据在所述分析中确定的唯一数据值的数量的更改,来确定包含在所述字段特定数据值列表的合并版本中的唯一数据值的数量;

-确定用于存储最高数量的数据记录ID的第一存储器要求,所述最高数量的数据记录ID在所述分析中确定并且要分配给所述字段特定数据值列表的合并版本中的数据值,以及确定用于存储要存储在所述字段特定数据值列表的合并版本中的最大数据值的第二存储器要求;

-根据所确定的列表中的唯一数据值的数量以及所述第一存储器要求和所述第二存储器要求,计算存储字段相关数据值列表的合并版本所需的存储器要求;

-确定物理数据载体上的邻接区域,所述邻接区域至少与计算出的存储器区域一样大;

-生成所述字段相关数据值列表的合并版本,所述合并版本集成自最后合并时间点以来存储在所述肯定列表和所述否定列表中的更改;以及

-将所述字段相关数据值列表的合并版本存储在所确定的所述数据载体的邻接区域中。

7. 根据前述权利要求中任一项所述的计算机实现方法,还包括提供所述字段特定数据值列表,其中,所述提供包括:

-解析原始数据以创建原始数据记录,其中,除了数据记录ID之外,每个原始数据记录还包括分配给其的一对或更多对字段标识符和原始数据值;

-在所述数据库中存储无冗余字段特定原始数据值列表,其中,所述无冗余字段特定原始数据值表之一中的原始数据值中的每一个已经向其分配了在表示所述原始数据值列表的字段中包含该原始数据值的那些原始数据记录的所有数据记录ID;

-生成映射表(210),所述映射表(210)向所述无冗余原始数据值列表的所述原始数据

值中的每一个分配未分配给任何其他原始数据值的至少一个映射ID;

-将所述原始数据记录转换为多个逻辑数据记录,并将无冗余字段特定原始数据值列表转换为所述字段特定数据值列表,其中,所述转换包括用根据所述映射表的映射ID替换原始数据值,其中,分配给所述数据记录的字段的数据值是所述映射ID。

8. 根据权利要求7所述的计算机实现方法,

-其中,所述映射ID是值,并且特别是数值,所述值的长度和/或类型优选地根据用于数据库搜索的计算机系统的处理器架构来选择。

9. 根据前述权利要求中任一项所述的计算机实现方法,其中,所述更改指令中的至少一个是更改或删除所述逻辑数据记录中的至少一个中的字段的废弃数据值的指令,其中,所述方法还包括:

-将至少一个数据记录的数据记录ID存储在被称为否定列表(220)的数据记录ID的列表中,其中,所述否定列表被链接到一个字段的字段标识符,并且被存储在链接到根据所述更改请求要被更改或删除的数据值的数据结构(216)中;

-其中,针对所述字段特定搜索值中的每一个执行所述数据库查询包括:

• 检查所述数据结构(216)是否包含存储为链接到数据值和字段标识符的否定列表,所述字段标识符和所述数据值与所述搜索值的字段标识符和字段特定搜索值相同;

• 如果是,计算在步骤i)中为该字段特定搜索值确定的所有数据记录ID与所述否定列表中的数据记录ID的差异量;以及

• 将数据记录ID的差异量用于步骤ii)-步骤iv。

10. 根据前述权利要求中任一项所述的计算机实现方法,其中,所述更改指令中的至少一个是将新数据值分配给所述数据记录中的至少一个中的字段的指令,其中,所述方法还包括:

-将至少一个数据记录的所述数据记录ID存储在被称为肯定列表的数据记录ID的列表中,其中,所述肯定列表存储在链接到所述一个字段的字段标识符并链接到新数据值的数据结构(216)中;

-如果所述新数据值替换了废弃数据值,则将所述至少一个数据记录的数据记录ID存储在被称为否定列表(220)的数据记录ID的列表中,其中,所述否定列表被存储为链接到所述一个字段的字段标识符并链接到所述废弃数据值;

-其中,针对所述字段特定搜索值中的每一个执行所述数据库查询包括:

• 检查所述数据结构是否包含被存储为链接到数据值和字段标识符的肯定列表,所述字段标识符和所述数据值与所述搜索值的字段标识符和所述字段特定搜索值相同;

• 如果是,则为所述字段特定搜索值计算在步骤i)中确定的所有数据记录ID与所述肯定列表中的数据记录ID的并集;其中,如果所述新数据值替换了废弃数据值,则所述并集的数据记录ID减去所述否定列表中分配给所述搜索值及其字段的数据记录ID,以及

• 将数据记录ID的并集用于步骤ii)-步骤iv。

11. 根据权利要求7-10中任一项所述的计算机实现方法,

-其中,所述数据结构(216)包含可搜索的、有序元素阵列,其中,所述阵列是列表元素的列表或节点的搜索树,特别是B树,

-其中,所述阵列在每种情况下表示所述字段中的一个,

-其中,所述阵列的元素各自表示包含在所述数据记录(214)中并分配给由所述阵列表示的字段的数据值的非冗余列表(116)中的一个数据值;以及

-其中,所述阵列的元素中的每一个被存储为链接到空的或非空的肯定列表和/或空的或非空的否定列表。

12.根据前述权利要求中任一项所述的计算机实现方法,

-其中,所述字段特定数据值列表(116)各自都是非冗余数据值列表,所述非冗余数据值列表选择向在所述逻辑数据记录中表示所述字段特定数据值列表的字段分配的那些数据值,

-其中,各个字段特定数据值列表中的每个数据值是唯一的,并且被存储为链接到在由所述字段特定数据值列表表示的字段中包含该数据值的所有逻辑数据记录的数据记录ID,

-其中,所述数据值优选地以有序形式存储在所述字段特定数据值列表中。

13.根据前述权利要求3-12中任一项所述的计算机实现方法,其中,所述合并包括:

-在所述第二时间点处,接收对自所述最近合并时间点以来指示的更改进行合并的命令;

-响应于接收到所述命令:

- 实施在所述最近合并时间点和所述第二时间点之间指示的所述字段特定数据值列表或所述字段特定数据值列表的副本的更改,以便生成合并字段特定数据值列表(228),使得即使在考虑了在所述第一时间点和所述第二时间点之间指示的字段中的更改之后,所述合并字段特定数据值列表之一的每个数据值仅被分配包含该数据值的那些逻辑数据记录的ID;

- 在所述第二时间点之后使用所述合并字段特定数据值列表而不是先前使用的字段特定数据值列表(216)来执行数据库搜索;

- 使用所述第二时间点作为新的最近合并时间点。

14.根据权利要求13结合权利要求9和/或权利要求10所述的计算机实现方法,还包括:响应于接收到所述命令并且在生成所述合并字段特定数据值列表之后:

-基于所述合并字段特定数据值列表重新生成至少一个数据结构(216),其中,所述数据结构的重新生成包括清空所述肯定列表和/或所述否定列表;

其中,在所述合并的过程中在所述最近合并时间点和所述第二时间点之间指示的更改的实施优选地包括通过分析受更改影响的所有字段特定数据值列表中的所有数据值的肯定列表和否定列表,来确定在所述最近合并时间点和所述第二合并时间点之间指示的更改。

15.根据前述权利要求中任一项所述的计算机实现方法,

-其中,所述逻辑数据记录中的至少一些包括一个或更多个“上级”字段,其中,所述“上级”字段中的每一个被配置为存储从属于所述数据记录的数据记录的数据记录ID;

-其中,所述字段特定数据值列表包括表示所述“下级”字段的数据值列表(416),其中,存储在所述数据值列表中的数据值是从属于至少一个其他逻辑数据记录的逻辑数据记录的ID,其中,“上级”数据值列表中的数据值中的每一个被分配其他下级数据记录的一个或更多个ID;

-其中,所述数据库查询包含完整性搜索参数,所述完整性搜索参数指定除了与所述数

数据库查询中确定的所述数据记录之外,是否还要输出从属于这些数据记录的数据记录;

-其中,所述数据库查询的执行包括:

- 确定所述下级数据记录也要被输出;
- 用在步骤i)中确定的所述数据记录ID搜索“上级”数据值列表,以获得从属于在步骤i)中确定的所述数据记录的数据记录的一个或更多个ID;
- 评估所述地址分配表,以识别向所确定的下级数据记录之一的ID分配的AOD条目的地址,
- 访问所述AOD条目的这些识别出的地址,以便将所述下级数据记录添加到所述数据库查询中确定的数据记录中。

16.根据前述权利要求中任一项所述的计算机实现方法,

-其中,所述字段特定数据值列表包括被称为时间值列表的多个字段特定数据值列表,其中,每个时间值列表由时间点的非冗余列表组成,其中,所述时间点各自表示该时间值列表所涉及的字段的数据值的有效性在所述逻辑数据记录中的一个或更多个中的开始或结束的时间点,

-其中,所述时间值列表中的所述时间点中的每一个被分配在所述时间点处有效的逻辑数据记录的ID;

其中,所述方法包括:

-响应于接收到与所述逻辑数据记录之一相关的更改请求,生成要更改的数据记录的新版本,其中,所述新版本是包含至少一个预版本字段的新逻辑数据记录,其中,所述预版本字段包含要更改的数据记录的ID,其中,所述新版本而不是要更改的所述数据记录包含所述更改请求中指定的更改和更改时间;以及

-将具有新数据记录ID的数据记录的新版本存储在所述字段特定数据值列表中,并存储所述时间值列表的新数据记录的有效性的开始以及所述更改指令所涉及的字段,其中,所述新版本的ID存储在所述时间值列表中;

-其中,优选地,所述数据库查询包含字段特定有效性时间的指示,其中,所述有效性时间指定在所述数据库查询中要确定的数据记录版本在对应字段中包含所述至少一个字段特定搜索值的时间点或时间段;并且其中,所述数据库查询的执行包括:

- 识别与所述搜索值和所述有效性时间相关的字段所关联的时间值列表;
- 用所述有效性时间搜索识别出的时间值列表,以便识别存储在与所述有效性时间或有效性时间段内的时间点相链接的所述时间值列表中的那些数据记录版本的一个或更多个数据记录ID;
- 识别与所述搜索值和所述有效性时间相关的字段所关联的所述字段特定数据值列表;
- 用所述搜索值搜索识别出的数据值列表,以便识别存储在与所述搜索值相链接的所述数据值列表中的一个或更多个数据记录ID;
- 计算基于识别出的时间值列表和识别出的数据值列表确定的所述数据记录的版本的ID的交集;
- 评估所述地址分配表,以识别向在先前步骤中确定的数据记录版本之一的ID分配的AOD条目的地址;

• 访问在先前步骤中识别的所述AOD条目的地址,以便通过其字段值补充在有效性时间点处或在有效性时间段期间有效的所述数据记录版本并输出该数据记录版本。

17. 根据前述权利要求中任一项所述的计算机实现方法,

- 其中,所述AOD条目被存储为所述仅添加数据结构中的区块链的元素,所述元素经由密码散列值链接在一起,以及

- 其中,所述数据库搜索的执行包括对在所述数据库查询的过程中处理的那些AOD条目的散列值的有效性检查。

18. 根据前述权利要求中任一项所述的计算机实现方法,其中,所述数据库查询由数据处理和搜索系统,即DMS系统(102)执行,所述DMS系统(102)被配置为以所述字段特定数据值列表的形式来管理和保持所述逻辑数据记录,其中,所述DMS系统被配置为除了所述字段特定数据值列表之外不管理和保持任何其他数据结构,其中根据所述任何其他数据结构,在第一合并时间处已经由所述DMS系统管理的所有逻辑数据记录还能够被重构。

19. 根据前述权利要求中任一项所述的计算机实现方法,其中,至少在执行步骤i)期间,所述字段特定数据值列表没有锁定于,即“没有数据库级锁定于”在步骤i)中搜索的任何或全部字段特定数据值列表,并且没有锁定于其中包含的数据值。

20. 一种易失性或非易失性存储介质,所述易失性或非易失性存储介质上存储有计算机可读指令,其中,所述指令被配置为使处理器执行根据前述权利要求中任一项所述的用于在数据库内执行数据库查询的方法。

21. 一种数据结构,包括:

- 多个字段特定数据值列表(116),其中以分布式方式存储逻辑数据记录,其中,每个数据记录包含数据记录ID和一个或多个字段标识符-数据值对,其中,优选地,每个字段特定数据值列表是仅分配给由所述逻辑数据记录中的列表表示的字段之一的那些数据值的有序无冗余列表,其中,字段特定列表中的每个数据值被存储为与包含该字段中的该数据值的所有数据记录的ID相链接;

- 仅添加数据结构(202),包含用于更改所述数据记录的字段的数据值的指令,其中,所述仅添加数据结构中的每个条目在此被称为AOD条目并且至少包含根据更改指令之一而要更改的所述数据记录之一的字段标识符-数据值对中的那些条目;

- 地址分配表(226),其中,所述地址分配表向以下数据记录中的每一个的ID分配对该数据记录的更改进行指定的最近AOD条目的地址(206、208):针对所述数据记录,具有更改指令的AOD条目被存储在所述仅添加数据结构中;

其中,所述数据结构尤其还包括:

- 至少一个数据结构(216),

• 其中,所述数据结构(216)包含可搜索的、有序的元素阵列,其中,所述阵列是列表元素的列表或节点的搜索树,特别是B树,

• 其中,所述阵列在每种情况下表示所述字段中的一个,

• 其中,所述阵列的元素各自表示包含在所述数据记录(214)中并分配给由所述阵列表示的字段的数据值的非冗余列表(116)中的一个数据值;以及

• 其中,所述阵列的元素中的每一个被存储为与空的或非空的肯定列表和/或空的或非空的否定列表相链接;和/或

- 从存储所述逻辑数据记录的原始数据获得的原始数据值的无冗余列表;和/或
 - 映射表(210),
 - 其中,所述映射表向所述原始数据值中的每一个分配未分配给任何其他原始数据值的至少一个映射ID;
 - 其中,所述数据记录的数据值是所述映射ID。
22. 一种计算机系统(100、500),包括:
- 至少一个处理器(108);
 - 数据存储器,包括数据库(104),其中,所述数据库在被称为“最近合并时间点”的第一时间点处包含多个逻辑数据记录,其中,每个数据记录包含数据记录ID和一个或多个字段标识符-数据值对,其中,每个数据记录包含数据记录ID和一个或多个字段标识符-数据值对,其中,所述数据记录以字段特定数据值列表的形式物理地存储(116);
 - 数据处理和搜索系统,即DMS系统(102),其中,所述DMS系统被配置为管理所述数据库(104),其中,在所述最近合并时间点之后的管理包括:
 - 接收更改所述数据记录中的若干数据记录的字段的数据值的指令;
 - 在不对所述字段特定数据值列表(116)进行更改的情况下,将所述指令存储(604)在仅添加数据结构(202)中,其中,所述仅添加数据结构中的每个条目在此被称为AOD条目并且至少包含根据更改指令之一而要更改的所述数据记录之一的字段标识符-数据值对中的那些条目;
 - 对于以下数据记录中的每一个:即关于该数据记录所述数据库在所述最近合并时间点之后接收到更改数据值的一个或多个指令,在所述地址分配表(226)中存储(606)对该数据记录的更改进行指定的被存储AOD条目中的最近AOD条目的地址(206、208),所述数据记录链接到所述数据记录的数据记录ID,其中,所述地址分配表中的链接被自动更新;以及
 - 执行(610)数据库查询,其中,所述数据库查询包括:
 - i. 基于与一个或多个字段特定搜索值的匹配,搜索(612)所述字段特定数据值列表以识别内容要被全部或部分返回的数据记录(214)的ID;
 - ii. 评估(614)所述地址分配表,以便识别分配给i)中识别出的数据记录ID之一的AOD条目的地址,
 - iii. 访问(616)识别出的AOD条目的地址;以及
 - iv. 使用(616、618)包含在这些识别出的AOD条目中的更改细节,以将字段标识符-数据值对添加到在步骤i)中确定的所述数据记录ID,并输出后者。

具有仅添加数据库结构的基于列表的数据搜索

技术领域

[0001] 本发明涉及一种用于存储数据并在数据内执行有效搜索的方法和系统。

背景技术

[0002] 在现有技术中,用于存储、管理和处理数据的各种数据库管理系统(database management system,DBMS)是已知的。DBMS的主要任务是高效、无矛盾并且永久地存储大量数据,并以不同的、基于需求的表示形式为用户和应用程序提供所需的子集。在由传统DBMS管理的数据库中用于构建数据及其相互关系的基础是数据库模型,该数据库模型由DBMS制造商限定。根据数据库模型,数据库模式(schema)必须适于特定的结构化选项。目前使用的众所周知的数据库模型包括层次结构模型、类网络模型、(按表格组织的)关系模型、面向对象的模型、面向文档的模型以及这些模型的混合形式。此外,通常在为有效回答许多小型查询(OLTP)或长期评估(OLAP)而优化的DBMS之间进行区分。

[0003] 处理大数据时的主要问题是资源问题。数据量越大,需要的存储器、处理器和硬盘驱动器形式的资源就越多。

[0004] 在创建数据库并限定要对数据库执行的查询和分析时,在支持的数据库搜索查询的复杂性和搜索速度之间往往存在目标冲突。的确,有一些DBMS在考虑时间、基于内容的和/或结构化的搜索标准的情况下允许进行复杂的搜索查询。通常,这些DBMS具有模块(“查询规划器”),该模块可以自动规划和策划对在多个表上复杂的、嵌套的查询的临时执行以及对从多个表中获取的部分结果的聚合。然而,对于非常复杂的查询和/或对于大量的表,这些模块很快就达到了它们的极限。在许多“大数据”应用中,要处理的和要加载到只读存储器(Read-Only Memory,RAM)中的数据量可能是巨大的,并且在实践中可能使评估复杂结构的大数据记录变得不可能。至少,通常不可能执行手头任务所需的所有形式的数据分析。特别受当前DBMS的限制影响的领域包括例如物联网(Internet-of-Things,IOT)、基因组分析、天文学领域中的光谱数据分析、来自出租车、公共汽车和铁路公司、航空公司和移动电话公司的移动数据的领域以及许多其他领域。

[0005] 通过在搜索标准和要输出的结果两者方面将所支持的搜索查询限制为较低复杂度的查询,传统数据库系统的速度通常可以被提高到有限的程度。然而,在一方面,这会导致无法对大数据记录进行重要分析,并且在许多情况下甚至可能导致不可逆转的数据丢失,因为原始数据被映射到复杂性降低的表结构,无法映射到这些表结构中的复杂的相互关系和元数据将丢失。

[0006] 因此,尤其是在处理与具有大量不同属性(键)和相应值的大量数据对象相关的复杂查询时,现有的DBMS通常以结构不灵活、可扩展性差和/或性能差为特点。如果随后必须在数据库中存储额外的、结构不同的数据,则通常不可能在内容方面可取并且在性能和资源方面有利的方式同时查询和分析现有数据和新添加的数据。

[0007] 大数据支持复杂分析和数据查询的挑战仍然被认为是未解决的;考虑到未来几年数据量的进一步增长,这个问题预计将变得更大。

发明内容

[0008] 本发明的目的是提供一种改进的用于存储数据和执行数据库查询的方法和系统，使得它们不具有上述问题或在较小程度上具有上述问题。

[0009] 形成本发明的基础的目的各自利用独立权利要求的特征来实现。在从属权利要求中描述了本发明的实施例。下面列出的实施例可以彼此自由地组合，只要它们不相互排斥。

[0010] 在一方面，本发明涉及一种用于在数据库内执行数据库查询的计算机实现方法。数据库在第一时间点(被称为“最近合并时间点”)处包含多个逻辑数据记录。每个逻辑数据记录包含数据记录ID和一个或多个字段标识符-数据值对。数据记录以字段特定数据值列表的形式物理地存储。该方法包括：在最近合并时间点之后：

[0011] -接收更改数据记录中若干数据记录的字段的数据值的指令；

[0012] -在不更改字段特定数据值列表的情况下，将指令存储在仅添加数据结构(append-only data structure)中，其中，仅添加数据结构中的每个条目(此处被称为AOD条目)至少包含根据更改指令之一要更改的数据记录之一的字段标识符-数据值对中的那些条目；

[0013] -对于数据库在最近合并时间点之后接收到的更改数据值的一个或多个指令的数据记录中的每一个，在地址分配表中存储对该数据记录的更改进行指定的被存储AOD条目中的最近AOD条目的地址，该数据记录链接到该数据记录的数据记录ID，其中，地址分配中的链接正在被自动更新；以及

[0014] -执行数据库查询，其中，数据库查询包括：

[0015] i. 基于与一个或多个字段特定搜索值的匹配，搜索字段特定数据值列表，以识别内容要被全部或部分返回的数据记录的ID；

[0016] ii. 访问地址分配表，以识别向在i)中识别出的数据记录ID之一分配的AOD条目的地址，

[0017] iii. 访问识别出的AOD条目的地址；以及

[0018] iv. 使用包含在这些识别出的AOD条目中的更改细节，以将字段标识符-数据值对添加到在步骤i)中确定的数据记录ID，并输出后者。

[0019] 这可能是有利的，有以下几个原因：

[0020] 数据库查询在两个不同的数据结构上执行，使得即使是最复杂的分析查询也可以被高速执行，并且可以快速输出完整的结果数据记录。首先，对字段特定列表执行步骤i)中的数据库查询。该步骤最初仅用于基于与一个或多个字段特定搜索值的匹配来识别要全部或部分返回内容的数据记录的ID。通常，启动了数据库查询的软件或用户不仅需要标识符，还需要首先从内容方面表示数据记录的属性(字段特定数据值)。在现有技术中，数据库查询通常已经被公式化，使得它在一个处理步骤中确定并返回要返回的数据记录，包括它们的属性。数据库查询也相应地复杂，通常在多个表上扩展，并包含各种连结(JOIN)操作。由于传统数据库查询的复杂性，经常需要加载和评估若干个表，并且数据库查询的性能较低，或者在一定复杂性以上是不可能的。然而，根据本发明实施例，对数据记录的属性的物理载体(在这种情况下是字段特定数据值列表)的分析仅用于确定要返回的数据记录ID。它不用于确定这些结果数据记录的属性。根据本发明实施例，要返回的数据记录的字段相关数据值不通过分析字段特定数据值列表来确定，而通过使用两种特定的数据结构(即仅添

加数据结构和地址分配表)来确定。

[0021] 仅添加数据结构是由几个按顺序更新的条目组成的数据结构。不可能对个别AOD条目进行后续删除或修改或重组。每个AOD条目指定对逻辑数据记录的一个或多个字段的一个或多个数据值的更改。

[0022] 这种仅添加数据结构在功能上与地址分配表紧密耦接:即,对于逻辑数据记录中的每个逻辑数据记录,地址分配表都包含一行该数据记录的ID。该数据记录的ID存储在地址分配表中,该地址分配表链接到恰好一个地址,其中,该一个地址是指定对该数据记录的更改的最近AOD条目的地址。

[0023] 为了现在向在步骤i)中确定的数据记录提供当前数据值并输出它们,管理数据库的基于软件和/或硬件的数据管理和搜索系统(DMS系统)首先访问地址分配表。

[0024] 根据本发明实施例,对于逻辑数据记录中的每个逻辑数据记录,地址分配表恰好包含具有该数据记录的ID的一个条目(例如恰好对应于表中的一行)。每个逻辑数据记录的ID存储在地址分配表中,该地址分配表恰好链接到AOD条目的一个地址,其中,该一个地址是指定对该数据记录的更改的最近AOD条目的地址。

[0025] 根据本发明的一些实施例,对地址分配表进行访问以确定地址分配表中与特定逻辑数据记录的ID相关的条目(即,例如,与要返回的数据记录的ID相关的条目)的实现方式如下:DMS系统被配置为生成由DMS系统管理的逻辑数据记录的ID,并以分布式方式存储在列表中,使得每个数据记录ID显式地或隐式地指定与该数据记录相关联的一个条目(“行”)在地址分配表中具有的偏移量。特别地,数据记录ID可以是数值。地址分配表中的条目的偏移量的明确规范可以是,例如,地址分配表中的条目的存储器地址相对于地址分配表的基本地址的规范。例如,隐式规范可以是地址分配表中该条目的编号(“行号”)的规范,其中,该条目编号与地址分配表中每个条目的预定义存储器大小的乘积给出该条目相对于地址分配表的基本地址的总偏移量。

[0026] 例如,DMS系统被配置为生成地址分配表,使得地址分配表内的条目(行)的位置不变,并且这些条目的偏移量与该条目相关联的逻辑数据记录的ID相同。当访问地址分配表时,DMS系统可以被适当地配置为确定特定逻辑数据记录的一个条目,以基于逻辑数据记录ID来计算该条目的位置,例如通过将作为条目的偏移量的数据记录ID乘以地址分配表中每个条目的指定的存储器位置大小。由此获得的乘积指示从地址分配表的基本地址开始的该一个数据记录的地址分配表中的条目的位置。

[0027] 例如,当生成新的逻辑数据记录时,DMS系统被配置为生成ID,并且还生成地址分配表中的新条目/新行,使得地址分配表中的该新条目的偏移量与新逻辑数据记录的ID相同。例如,逻辑ID可以是数字数据值或存储器地址。

[0028] 例如,地址分配表的基本地址可以具有存储器地址#23862382844。每一行可能需要例如16字节的预定义存储器空间。然后,DMS系统可以如下计算具有例如ID 1008的逻辑数据记录的条目的存储器地址:基本地址+(地址分配表条目的存储器大小×逻辑数据记录的ID)=#23862382844+16字节×1008。计算出的存储器地址在这里被虚拟地称为#23862992849。

[0029] 根据另一个明确的实现变体,数据记录的逻辑ID将与地址分配表中相应条目的存储器地址相同,也就是说,例如#23862992849,而不是上述示例中的“1008”。

[0030] 因此,根据本发明实施例,DMS系统对逻辑数据记录的管理和对地址分配表中的条目的管理紧密耦接,使得逻辑数据记录总是被生成并且与地址分配表同步,使得每个逻辑数据记录的ID显式或隐式地指定地址分配表中一个条目的存储器地址,其中该逻辑数据记录的ID被分配具有该数据记录的最近更改的AOD条目的地址。根据实施例,当处理数据库查询时,要返回的数据记录的逻辑ID用于确定地址分配表中与该数据记录相关联的一个条目的地址,并直接访问该地址。

[0031] 这种确定地址分配表中包含要输出的数据记录的ID的行的方法需要很小的RAM和很小的中央处理单元(Central Processing Unit,CPU)功率,因为DMS系统可以立即且直接地访问地址分配表中的相应条目,或者,如果需要,在简单且非常快的可执行乘法之后访问地址分配表中的相应条目。没有必要搜索地址分配表。从地址分配表中的该条目开始,DMS系统可以直接访问AOD数据结构中的相关条目,因为具有最近更改的该AOD条目的地址在地址分配表的先前确定的条目中被指定。如果有更多的AOD条目指定了对该数据记录的进一步的、较旧的更改,则也可以通过直接存储器访问来非常快速地确定和读取这些更改,因为优选地,每个AOD条目都包含最近AOD条目(至少如果AOD条目不是“完全加载的”条目,即不包含相应数据记录的所有当前字段值)的地址。

[0032] 因此,DMS系统只需从地址跳到地址,以输出数据记录作为任何类型的数据库查询的结果,而不必在多个表之间执行复杂的JOIN或选择(SELECTS)。这可以导致即使是最复杂的数据库查询也会大大加速,因为一旦确定了数据记录的ID—这可以在无冗余列表的基础上非常快速地完成—完整数据记录的输出基本上或完全基于跳转地址运行,即在数据结构中没有顺序搜索和/或表的复杂JOIN操作。

[0033] 例如,该AOD条目可以包含所述数据记录的所有字段的所有当前数据值,使得通过对该一个AOD元素的访问,可以已经确定并返回该数据记录的所有当前数据值。

[0034] AOD条目也可能只包含该数据记录的所有字段的数据值中的一个或更多个数据值(而不是全部数据值)中的当前数据值(本文也被称为“不完整AOD条目”)。然而,根据本发明实施例,“不完整”AOD条目包括指向引用相同数据记录的下一个较旧AOD条目的地址。该地址(也被称为跳转地址)允许DMS系统经由AOD条目中指定的相应地址直接从AOD文件中的AOD条目“跳转”到AOD条目,从而重建影响该特定数据记录的所有更改。这可以进行到已经为数据记录的所有字段确定了当前数据值为止。

[0035] 因此,根据本发明实施例,从在步骤i)中确定的数据记录ID开始,可以高效且快速地确定这些数据记录的字段值,只需要传统DBMS所需的CPU和/或RAM资源中的一小部分来确定和输出复杂的结果数据记录,因为从来自步骤i)的数据记录ID开始,只需要确定地址分配表中基于ID的查询结果(hit),并且从这些开始,特别需要读出一个或更多个AOD条目的一个或更多个存储器地址。这是非常有效的,因为例如,不需要仅添加数据结构的线性扫描,而是可以直接对由地址指定的AOD条目进行读取访问。

[0036] 根据本发明实施例,字段特定数据值列表是无冗余的、排序的字段特定数据值列表,其中每个数据值在每个列表中仅出现一次。

[0037] 根据实施例,数据库查询由数据处理和搜索系统(DMS系统)执行,该系统管理逻辑数据记录并以字段特定数据值列表的形式保存它们。除了字段特定数据值列表之外,DMS系统不包含任何另外的的数据结构,根据这些数据结构可以重建在第一合并时间点处已经由

DMS系统管理的所有逻辑数据记录。尽管DMS系统可以临时解析或导入原始数据以读取数据记录,但这仅用于数据记录的提取。因此,数据记录不会额外地永久存储(保存)在表或其他数据结构中。

[0038] 这可能是有利的,因为可以以只需要非常小的存储空间的方式存储非常大量的数据。在传统数据库中,数据首先存储在主要或专门用于保存数据的数据结构中,也就是说,例如存储在表或在图数据库的节点中。除了这些结构(取决于所使用的DBMS的类型和配置)之外,还会创建诸如搜索索引之类的附加结构,以实现快速搜索。因此,数据以冗余方式存储在表和索引两者中。然而,根据实施例,逻辑数据记录仅以字段特定数据值列表中的条目的形式存储。本文描述的附加数据结构(诸如AOD结构、肯定列表(positive list)和/或否定列表(negative list))特别用于存储在下次合并期间集成到现有字段特定数据值列表或现有字段特定数据列表的副本中的更改信息。根据本发明实施例,字段相关列表因此既用于永久存储逻辑数据记录,又用于实现快速搜索。

[0039] 据观察,通常情况下,这种类型的逻辑数据记录存储不仅避免了以附加搜索索引的形式存储附加数据,而且事实上,压缩通常发生在原始数据的存储过程中:例如,具有包含整数、浮点数和字符串形式的天文测量值和参数的多个列的大小为1.5TB的CSV文件以1.3TB大小的字段特定数据值列表的形式存储。数据压缩之所以成为可能,是因为每个字段特定数据值在列表中仅出现一次。存储对其中发生数据值的逻辑数据记录的引用也会占用存储空间。由于引用以数据记录ID的形式存储,数据记录ID可以以节省空间的方式存储,例如作为预定义长度的数值,因此存储数据记录ID所需的附加存储器需求通常小于通过仅存储一次属于特定字段/参数的所有数据值而保存的存储器需求。根据实施例,可以通过在合并过程中确定字段特定数据值列表中每个字段特定数据值列表的合并版本的条目的数量和最小所需存储器大小,并根据该确定的结果来对合并的数据值列表的结构和/或其存储位置进行选择以实现进一步的节省。

[0040] 该数据库具有98个字段的18.5亿个数据记录。

[0041] 根据实施例使用的数据结构还允许极其高效的数据查询。例如,上面提到的1.3TB数据集包含98个字段的18.5亿个逻辑数据记录。在根据本发明实施例的数据结构的基础上,在标准计算机(516GB RAM、2个Intel Xeon CPU E5-2643 v4@3.40GHz)上,即使是涉及大量字段的非常复杂的查询也可以在几毫秒内(在某些情况下在几秒内)执行。

[0042] 字段特定数据值列表中的每个字段特定数据值列表的每个数据值都被分配了所有逻辑数据记录的数据记录ID,根据它们的字段标识符-数据值对,所有逻辑数据记录包含由该列表表示的字段的数据值。数据记录中的每个数据记录以分布式方式物理存储在一个或更多个字段特定数据值列表中(其中,分配给多个列表的数据值的相同数据记录ID将多个数据值组合成概念上一致的数据记录)。在步骤i)中对字段特定数据值列表进行搜索是通过将字段特定搜索项与那些字段特定数据值列表的排序数据值相匹配来执行的,其中所表示的字段与字段特定搜索项的字段相同。

[0043] 根据实施例,至少在执行步骤i)期间,字段特定数据值列表在步骤i)中搜索的任何或所有字段特定数据值列表上没有锁定(“数据库锁定”),并且在其中包含的数据值上没有锁定。这可以显著加快数据库查询速度。

[0044] 根据一个实施例,AOD条目的地址可以例如由仅添加数据结构的第一AOD条目的物

理地址与偏移量组合组成,其中,偏移量是AOD条目相对于仅添加数据结构的第一个AOD条目的地址。

[0045] 在传统的DBMS中,在确定要返回的数据记录集的数据记录ID和也要返回的这些数据记录的属性之间没有进行区分。相反,单个复杂的、通常基于结构化查询语言(Structured Query Language,SQL)的查询指定要分析哪些表以及要评估和/或返回那些表的哪些列。然而,本发明实施例允许数据库查询是两步查询,其中,只有对要返回的数据记录的ID的确定在数据的物理载体结构上执行,也就是说在这种情况下是字段值特定列表。要与数据记录的属性一起返回的数据记录的丰富性基于两种特定数据结构,即所描述的地址分配表和仅添加数据结构的条目,地址分配表的条目引用的仅添加数据结构的条目的地址。

[0046] 在另一个有利的方面,数据库也可以被非常有效地使用和合并。由于所请求的更改不是在第一时间点和稍后(“第二”)时间点之间立即存储在字段特定列表中,而是最初仅以AOD条目的形式存储在仅添加数据结构中(并且根据一些实施例,也存储在所谓的肯定列表和否定列表中),读取请求仍然可以在没有字段特定列表上的任何“锁定”的情况下执行。字段特定列表因此表示在第一时间点和第二时间点期间的静态(即,不变状态)。读取请求可以在这些字段特定列表上非常有效地执行,因为在某些读取访问之前,不需要临时锁定列表以确保查询的一致性。

[0047] 例如,DMS系统可以被配置为只允许在AOD数据结构上并行运行一个写入过程和大量读取过程。因此,在不因数据一致性原因引入锁定的情况下,可以从许多读取过程中评估连续补充的AOD数据结构的内容,以便完成各种数据库查询。这里不存在逻辑冲突,因为仅添加结构的性质是,补充数据结构所需的一个写入过程和几个读取过程不会相互干扰,特别是,两个写入过程不可能同时访问同一存储器。根据本发明实施例,该方法还包括通过独立于对字段特定数据值列表上的数据库查询的执行和/或与对字段特定数据值列表上的数据库查询的执行并行地合并字段特定数据值列表,来合并自最近合并时间点以来指示的更改。合并第二时间点(被称为新合并时间点)处执行。

[0048] 例如,字段特定数据库列表的合并可以包括使用基于根据实施例使用的肯定和/或否定列表存储的(涉及与字段相关数据值及其与逻辑数据记录的关联相关的)信息,来更新字段特定数据库列表的数据值和数据记录ID。

[0049] 根据实施例,该方法还包括,对于字段特定数据值列表中的每个字段特定数据值列表的每个数据值:

[0050] -在被称为否定列表的列表和数据记录ID中,存储由于在第一合并时间点和第二合并时间点之间接收到的更改指令而不再被分配关于该字段的该数据值的所有逻辑数据记录的数据记录ID;以及

[0051] -在被称为肯定列表(218)的列表和数据记录ID中,存储由于在第一合并时间点和第二合并时间点之间接收到的更改指令而要被分配关于该字段的该数据值的所有逻辑数据记录的数据记录ID。

[0052] 执行合并包括:根据肯定列表和否定列表的内容,来更新字段特定数据值列表或字段特定数据值列表的副本的数据值和数据记录ID。合并之后,肯定列表和否定列表将被清空。

[0053] 根据实施例,该方法包括与合并并行地且独立地、不中断地对字段特定数据值列表执行进一步的数据库查询。附加地或替代地,字段特定数据值列表在合并的执行期间也不锁定于包含在其中的数据值。

[0054] 所描述的合并可能是有利的,因为将数据记录的字段特定数据值存储在字段特定数据值列表中确保在合并期间只有少数(可能没有)列表元素需要被临时锁定以禁止访问,以避免不一致的查询结果。例如,如果在合并数据值列表的过程中添加了与字段相关的全新数据值,也就是说,例如,在特定于链接到包含该值的数据记录的ID的字段“名字”的数据值列表中存储了以前未知的名字“Torben”,该列表只包含一个新条目,并且不必锁定以前的条目。如果数据记录的现有数据值发生了更改,也就是说,例如,在特定数据记录中,包含拼写错误的名字“Micchael”被更正为“Michael”,则数据值“Michael”的输入不会更改数据值本身,而是仅分配给该数据值的数据记录ID由现在更正的数据记录的ID补充。因此,在该合并期间,仍可以分析和处理名字列表的数据值。

[0055] 对于传统的DBMS,只要它们支持利用高速缓存的更改来合并数据库,由于具有多个依赖项和逻辑约束的复杂表结构,通常需要在合并期间完全锁定多个列甚至表,以确保查询结果的逻辑一致性。例如,在传统DBMS的写入和合并过程中,经常使用基于信号量机制的锁定,这可能会显著降低系统在处理数据库查询方面的性能。根据本发明实施例,这是不必要的,因为在一方面,与地址分配表和仅添加数据结构相结合的字段特定列表结构使得复杂的表结构和约束在数据库设计中是不必要的,并且因为在另一方面,在许多内容相关更改的情况下,受影响的不是数据值,而是分配给各个字段特定数据值列表中的这些数据值的数据记录ID的类型和数量。例如,约束可以是变量的值必须满足的条件,以便该值被接受到表中。此外,根据本发明实施例,最初根本不在字段相关数据值列表中进行更改,而是在AOD数据结构、肯定列表和否定列表中进行更改,使得数据值列表可以不受限制地用于处理数据库查询。只有在合并过程中,才必须在写入过程中访问数据值列表或其副本,但在大多数情况下,现有数据值根本不受更改的影响,而最多受存储的、链接到数据值的数据记录ID的影响。根据实施例,仅当在删除之后,数据值不再被分配给逻辑数据记录中的任何逻辑数据记录中的字段时,数据值才是被删除的数据值列表的相应数据值(但也可以将该数据值保留在数据值列表中,即使它不再出现在逻辑数据记录中,因为该数据值可能在以后链接到将来要存储的数据记录)。

[0056] 替代地,该方法还包括:通过独立于对字段特定数据值列表执行数据库查询和/或与对字段特定数据值列表执行数据库查询并行地生成字段特定数据值列表的合并副本,来合并自最近合并时间点以来指示的更改。合并的第二时间点(被称为新合并时间点)处执行。

[0057] 该实施例的优点对应于其中直接对已经针对数据库查询分析的数据值列表执行合并的实施例的优点,其中,该优点更进一步,因为可以完全避免锁定数据值或数据记录ID分配,因为合并对字段特定数据值列表的副本执行,而不是对用于当前搜索查询的字段特定数据值列表执行。

[0058] 例如,在合并过程中,特别是在后台中,特定数据库列表的合并副本的创建可以由DMS系统自动执行字段,而并行数据库查询由DMS系统在正常操作中使用AOD数据结构并且优选地还使用肯定列表和否定列表对尚未合并的数据值列表执行。DMS系统被配置为在合

并完成之后对字段特定数据值列表的合并副本执行数据库查询,而不是对未合并的字段特定数据值列表执行数据库查询。因此,这种“切换”导致从完成合并或切换时起对合并的数据值列表执行新的查询。

[0059] AOD数据结构、肯定列表和否定列表因此允许执行返回当前结果的数据库查询,即使字段特定数据值列表本身可能不再是最新的,同时在后台中对这些数据值列表的副本执行合并过程。这使得在后台中进行的合并期间,数据库查询的数据能够高性能且不间断地可用。

[0060] 根据本发明实施例,对于字段特定数据值列表中的每个字段特定数据值列表,合并包括:

[0061] -分析数据记录ID的字段特定和数据值特定的肯定列表和否定列表,以确定与字段的唯一数据值的数量相关的更改以及与第一合并时间点和第二合并时间点链接到该字段的数据值的数据记录ID的最高数量相关的更改;

[0062] -根据在分析中确定的唯一数据值的数量的更改,来确定包含在字段特定数据值列表的合并版本中的唯一数据值的数量;

[0063] -确定用于存储最高数量的数据记录ID的第一存储器要求,最高数量的数据记录ID在分析中确定并且要分配给字段特定数据值列表的合并版本中的数据值,以及确定用于存储要存储在字段特定数据值列表的合并版本中的最大数据值的第二存储器要求;

[0064] -根据所确定的列表中的唯一数据值的数量以及第一存储器要求和第二存储器要求,计算存储字段相关数据值列表的合并版本所需的存储器要求;

[0065] -确定物理数据载体上的邻接区域,该邻接区域至少与计算出的存储器区域一样大;

[0066] -生成字段相关数据值列表的合并版本,该合并版本集成自最后合并时间点以来存储在肯定列表和否定列表中的更改;以及

[0067] -将字段相关数据值列表的合并版本存储在所确定的数据载体的邻接区域中。特别地,存储可以使得数据值列表的每个条目具有相同的大小,例如至少与计算出的第一存储器要求和第二存储器要求之和一样大。

[0068] 例如,数据值列表“名字”的非合并版本可能包含312,234个不同的条目(名字)。基于对AOD数据结构的分析,DMS系统确定这些名字中的3个已经在数据集中被全局删除(也就是说,这3个名字的删除不仅影响现在已经被删除或具有不同名字的三个单独的数据记录,而且由于这三个删除或更改,不再提供具有这三个名字之一的单个逻辑数据记录)。作为回报,AOD结构中保存了数百个新的数据记录,包括具有两个全新名字的人的数据记录。因此,在合并名字数据值列表的过程中,必须从列表中删除三个名字,并添加两个新的名字。合并名字列表中的元素的数量增加了一个元素的净额,达到312,235个不同的条目(名字)。DMS系统可以计算合并名字列表所需的存储器要求,例如作为条目的数量(312,235)与单个条目的存储器要求的乘积。合并的数据值列表优选地以这样的方式存储,即该列表存储在连续的物理存储器区域中,该物理存储器区域至少与312,235个条目与数据值列表的每个条目所需的存储器容量的乘积一样大。

[0069] 例如,数据值列表的每个条目所需的存储器空间可以定义如下:为存储数据值本身而保留的存储器区域对应于该列表中最大或最长的数据值所需的空间。例如,最长的名

字可以是20个字符(表示为字符(char)数据类型,每个字符的存储器要求为8位),在这里导致每个名字的存储器区域值为160位。然而,第一名字列表中的每个条目还额外地包括一组逻辑数据记录ID,其例如可以表示为整数数据类型。DMS系统被配置为通过分析尚未合并的名字列表和AOD数据结构,来确定分配给名字的数据记录ID的最大数量。例如,一些名字(例如“Andreas”)经常出现,因此具有大量的数据记录ID,而其他名字(例如“XAE A-12”)则很少或唯一。最频繁出现的名字(考虑存储在AOD结构和/或名字的肯定列表和否定列表中的相关变化)决定了为每个条目的数据记录ID的量而保留的存储器区域的大小。例如,如果名字“Andreas”出现频率最高,特别是例如在502,344个逻辑数据记录中,并且如果每个数据记录ID都存储为32位整数,则名字数据值列表的每个条目必须保留至少 $502,344 \times 32 \text{位} = 16,075,008 \text{位}$ 的存储器来存储数据记录ID。因此,包括用于存储名字数据值的160位在内,必须为每个条目保留16,075,168位的存储器。优选地,对于字段(分配的数据记录ID的集合)的数据值观察到的最大出现频率被存储为链接到相应的数据值列表,并且在每次合并的过程中被新确定和更新。

[0070] 在这里描述的示例中,通过取312,235个条目与数据值列表的每个条目所需的16,075,168位的存储器容量的乘积,来计算存储名字列表的合并版本所需的连续物理存储器区域的大小。

[0071] 因此,根据实施例,在合并过程中合并的数据值列表的存储是以这样的方式进行的,即每个合并列表被存储在数据载体的连续物理区域中,并且被存储在预定大小(数据值本身以及链接到它的数据记录ID的集合的预定大小)的数据结构中。整个列表所需的存储器要求、数据值所需的大小以及链接的存储的数据记录ID的量两者都会(在必要时)针对每次合并进行重新计算和调整。这可能是有利的,因为,在一方面,如果列表的数据值在物理上彼此相邻地连续存储,并且读/写头不必进行任何跳跃,则可以特别快速地读出数据。在另一方面,数据也以非常节省空间的方式存储,因为数据值或数据记录ID的集合保留的存储器区域动态地适应实际存在的数据的特性。因此,用于存储的数据结构自动且灵活地适应要存储的数据,使得仅保留实际需要的存储器区域。在数据库的环境中,可变大小的数据值通常存储在单独的数据结构中,然后仅经由指针引用这些数据结构。但是,这种存储可能会对性能产生不利影响。根据实施例,每个数据值及其相关联的数据记录ID被存储在预定大小的数据结构中,其中,各个数据记录被连续地存储在物理数据载体上。根据一些实施例,DMS系统被配置为使用各个字段相关数据值列表的基本地址以及单个列表条目的偏移量和存储器大小,来指定和访问字段相关数据值列表的每个条目的物理存储器地址。这可以显著减少访问时间。通过使用预定大小的数据结构,基于偏移量的直接的、有针对性的访问成为可能。

[0072] 根据本发明实施例,该方法包括提供字段特定数据值列表,特别是初始提供字段特定数据值列表。该提供包括:

[0073] -解析原始数据以创建原始数据记录,其中,除了数据记录ID之外,每个原始数据记录还包括分配给其的一对或更多对字段标识符和原始数据值;

[0074] -在数据库中存储无冗余字段特定原始数据值列表,其中,无冗余字段特定原始数据值表之一中的原始数据值中的每一个已经向其分配了在表示原始数据值列表的字段中包含该原始数据值的那些原始数据记录的所有数据记录ID;

[0075] -生成映射表,该映射表向无冗余原始数据值列表的原始数据值中的每一个分配未分配给原始数据值中的任何其他原始数据值的至少一个映射ID;

[0076] -将原始数据记录转换为多个逻辑数据记录,并将无冗余原始数据值列表转换为无冗余字段特定数据值列表,其中,转换包括用根据映射表的映射ID替换原始数据值,其中,分配给数据记录的字段标识符的数据值是映射ID。

[0077] 例如,原始数据值可以是字符串,例如名字“Michael”,并且分配给该原始数据值的映射ID可以是诸如“3493487879”之类的数值。

[0078] 根据其他实施例,原始数据值可以直接用作逻辑数据值,并且字段特定列表在这种情况下是字段特定原始数据值列表。

[0079] 在生成和存储字段特定列表时使用映射ID而不是原始数据值,当根据字段特定数据值执行搜索和评估搜索值时,具有显著加快该方法并最小化资源消耗的优点:虽然例如在解析和/或标记化步骤中获得的原始数据值的长度是不可预测的,并且例如以字符串(string)或可变长度字符串(varchar)数据类型的形式存在的原始数据值的存储需要大量存储器空间,但是映射ID的长度可以固定为统一值。此外,映射ID可以被存储为数值,从而与统一码(Unicode)字符串相比,可以减少存储器消耗并且可以显著提高处理速度。

[0080] 此外,数据库因此变得更加安全,因为数据值列表现在仅由数字映射ID组成,这不允许在不知道映射表的情况下对原始数据值的重建。

[0081] 在下文中,为了解释逻辑数据值的字段特定数据值,通常使用其中数据值为原始数据值的示例,例如统一码字符串。然而,优选地,数据值是数字数据值,特别是映射ID,其被唯一地分配给映射表中的原始数据值。根据实施例,映射表可以包括几个字段特定映射表,或者由全局映射表组成,该全局映射表将映射ID分配给任何字段中存在的所有数据值。

[0082] 根据本发明实施例,映射ID是优选地根据用于数据库搜索的计算机系统的处理器架构来选择其长度和/或类型的值。特别地,映射ID可以是数值。

[0083] 例如,所有映射ID可以具有与所使用的处理器架构的寄存器长度相对应的特定长度(例如32位或64位)。这可以进一步提高数据处理的速度并减少资源消耗。

[0084] 根据本发明实施例,生成映射表包括分析原始数据记录中原始数据值的出现频率。如果原始数据记录中原始数据值的出现频率超过阈值,则该方法包括将多个不同的映射ID分配给映射表中的这一个原始数据值。在变换中替换原始数据值,使得一个原始数据值被多个映射ID替换,从而模糊一个原始数据值的原始出现频率。

[0085] 这可以具有这样的优点,即可以排除基于出现频率从原始数据值的映射ID进行的推断。根据语言或数据记录,某些原始数据值,例如某些单词,出现频率很高(在德语中,例如“und”、“ein”、“eine”或“das”(“and”、“a”或“the”)之类的词,而其他数据值很少出现(例如“Schiffschraube”(船的螺旋桨))。因此,特别是在数据库的内容大致已知的情况下,即使这些列表仅包含映射ID并且映射表未知,也可以从字段特定列表推断出原始数据记录有一定风险,因为数据值或映射ID的出现频率提供了原始数据值的线索。根据本发明实施例,这可以通过可能地将几个不同的映射ID分配给相同的数据值使得数据值在原始数据记录中的出现频率被模糊来避免。例如,原始数据值“Michael”最初可能被分配了映射ID“9237923”。当在特定字段中包含特定数据值(例如“Michael”)的原始数据记录的数量超过阈值(例如1,000或10,000等等,取决于应用)时,该数据值在每个后续数据记录中的出现被

存储在数据值列表和映射表中,从该数据值在数据记录中的下一次(例如第1,001次或第10,001次)出现起,新的映射ID“23747283472”被分配给数据值“Michael”,并且字段特定数据值列表也被该新的数据值与分配的数据记录ID一起扩展。

[0086] 根据本发明实施例,更改指令中的至少一个更改指令是更改或删除逻辑数据记录中的至少一个逻辑数据记录中的字段的废弃数据值的指令。该方法包括将至少一个数据记录的数据记录ID存储在被称为否定列表的数据记录ID的列表中。否定列表链接到一个字段的字段标识符,并且被存储在链接到根据更改请求要更改或删除的数据值。

[0087] 针对字段特定搜索值中的每个字段特定搜索值,数据库查询的执行包括:

[0088] -检查数据结构是否包含存储为链接到数据值和字段标识符的否定列表,字段标识符和数据值与搜索值的字段标识符和字段特地给你搜索值相同;

[0089] -如果是,则计算在步骤i)中为该字段特定搜索值确定的所有数据记录ID与否定列表中的数据记录ID的差异量;以及

[0090] -将数据记录ID的差异量用于步骤ii-步骤iv。

[0091] 这可能是有利的,因为它确保了由于自最近合并时间点以来一个或多个数据值的变化而不应再在数据库查询中找到的数据记录,但是由于数据值列表尚未反映这些变化而最初在步骤i)中找到的数据记录不被返回。例如,对于字段特定数据值列表中的每个字段特定数据值列表,可以在数据库中创建数据结构,该数据结构对于该数据值列表中包含的每个数据值包含在(无论是通过重写数据值还是通过删除数据值或通过删除整个数据记录)表示数据值列表的字段中删除了该值的数据值特定“否定列表”中的那些数据记录ID的列表。这确保数据库查询还考虑到尚未合并的特定字段的数据值的删除。

[0092] 根据本发明实施例,更改指令中的至少一个是将新数据值分配给数据记录中的至少一个中的字段的指令。该方法还包括:

[0093] -将至少一个数据记录的数据记录ID存储在被称为肯定列表的数据记录ID的列表中,其中,肯定列表存储在链接到一个字段的字段标识符并链接到新数据值的数据结构中;

[0094] -如果新数据值替换了废弃数据值,则将至少一个数据记录的数据记录ID存储在被称为否定列表的数据记录ID的列表中,其中,否定列表被存储为链接到一个字段的字段标识符并链接到废弃数据值。

[0095] 针对字段特定搜索值中的每一个,数据库查询的执行包括:

[0096] -检查数据结构是否包含被存储为链接到数据值和字段标识符的肯定列表,字段标识符和数据值与搜索值的字段标识符和字段特定搜索值相同;

[0097] -如果是,则为字段特地给你搜索值计算在步骤i)中确定的所有数据记录ID与肯定列表中的数据记录ID的并集;其中,如果新数据值替换了废弃数据值,则并集的数据记录ID减去否定列表中分配给该搜索值及其字段的数据记录ID,以及

[0098] -将数据记录ID的并集用于步骤ii-步骤iv。

[0099] 这可能是有利的,因为它确保了返回由于自最近合并时间点以来一个或多个数据值的变化而应该在数据库查询中被另外找到但是由于数据值列表还未反映该变化而在步骤i)中未被最初找到的数据记录。例如,对于每个字段特定数据值列表,可以在数据库中创建数据结构,该数据结构对于该数据值列表中包含的每个数据值,包含数据值特定“肯定列表”中的那些数据记录ID的列表,通过重写另一废弃数据值或通过第一次将数据值存储

在该数据记录的字段中或通过第一次存储整个数据记录,来将该值存储在表示数据值列表的字段中。这确保了数据库查询还考虑到尚未合并的特定字段或新写入数据记录的数据值的更改和补充。

[0100] 根据本发明实施例,数据结构包含可搜索的、有序元素阵列。该阵列是列表元素的列表或节点的搜索树。特别地,搜索树可以是B树。该阵列在每种情况下表示字段中的一个。阵列的元素表示包含在数据记录中并分配给由阵列表示的字段的数据值的非冗余列表中的一个数据值。阵列的元素中的每一个被存储为链接到空的或非空的肯定列表和/或空的或非空的否定列表。

[0101] 以排序的、可搜索的形式实现肯定列表和/或否定列表可能是有利的,因为这也允许在数据库查询中非常快速地考虑动态变化的和尚未合并的数据。

[0102] 到目前为止,传统数据库还不允许对尚未转移到实际用于存储的(表)形式中的动态的、未合并的数据的有效查询,因为这里同时出现了几个问题:尚未合并的例如存储在高速缓存中的新数据,不能被对用于数据存储的实际(表)结构进行操作的查询所捕获。此外,它们在高速缓存中的结构中可用,这些查询无论如何都无法在该结构上工作。相反,肯定列表和否定列表与所描述的其他数据结构(特别是字段特定数据值列表、仅添加数据结构和地址分配表)相结合的使用,具有对数据记录集合运算足以识别响应于数据库查询最终要输出的数据记录集合的优点:步骤i)和在否定列表和/或肯定列表中的搜索最终都返回可以通过对这些数据记录ID进行集合运算以高效方式处理的数据记录ID。

[0103] 例如,搜索查询可能包括相对于字段“名字”的搜索值“Michael”。在步骤i)中,用非冗余数据值列表中的搜索值搜索字段“名字”,会产生数据记录ID的初始集合,该数据记录ID涉及在最近合并时间点处在“名字”字段中具有值“Michael”的数据记录。

[0104] 对字段“名字”的数据值“Michael”的否定列表进行评估会产生数据记录ID的第二集合,这些数据记录ID涉及自最近合并时间点以来名字“Michael”已被删除(例如由于覆盖或因为整个数据记录已被删除)的数据记录。然而,由于在i)中确定的第一集合自最近合并时间点以来不包含任何变化,因此必须从第一集合的ID中减去第二集合的ID,即必须形成差异集合,以防止输出同时在名字字段中不再包含搜索值“Michael”的数据记录。

[0105] 对字段“名字”的数据值“Michael”的肯定列表进行评估会产生数据记录ID的第三集合,这些数据记录ID涉及自最近合并时间点以来(例如由于覆盖废弃值或因为新建了整个数据记录)添加了名字“Michael”的数据记录。然而,由于在i)中确定的第一集合自最近合并时间点以来不包含任何变化,因此必须将第三集合的ID添加到第一集合的ID,即必须由集合1和集合3形成并集,以确保仅在最近合并时间点之后才输出在名字字段中包含搜索值“Michael”的数据记录。

[0106] 因此,肯定列表和/或否定列表的使用使得可以非常快速地确定数据记录ID的第二集合和/或第三集合,该第二集合和/或第三集合可以通过与在步骤i)中确定的数据记录ID的集合的集合运算来聚合,以便确定要最终返回的数据记录的集合。在这里,也经由地址分配表和仅添加数据结构来丰富要输出的具有属性(数据值)的数据记录。由于在补充要输出的数据记录ID时可以使用AOD条目的跳转地址,因此不需要复杂的跨表JOIN或类似的复杂操作,因此输出非常高效。

[0107] 根据实施例,DMS可以仅对已经合并的数据和/或对包括动态(即尚未合并的)数据

的所有可用数据执行数据库查询。在任何情况下都不存在数据不一致或由于锁定而导致时间延迟的风险。

[0108] DMS系统可以仅对在最近合并时间点处合并的数据执行数据库查询,例如通过在步骤i)中仅使用数据值列表而不使用肯定列表或否定列表来确定数据记录ID,并且通过仅评估AOD条目来完成在i)中确定的数据记录并返回,其是经由地址分配表为这些数据记录ID确定的并且比最近合并时间点更早。

[0109] 附加地或替代地,DMS系统可以对可用数据的整体(即对合并的和尚未合并的数据)执行数据库查询,例如通过使用数据值列表以及另外的肯定列表和否定列表来确定步骤i)中的数据记录ID,并且通过评估所有AOD条目来完成在i)中确定的并且要返回的数据记录,这些数据记录是经由用于这些数据记录ID的地址分配表确定的。

[0110] 如果数据值是映射ID,则根据实施例,在输出之前用分配给映射表中的映射ID的原始数据值来替换数据值。

[0111] 根据本发明实施例,阵列的元素各自表示数据值,数据值是数值并且被称为映射ID。映射ID被分配给从映射表中的原始数据获得的恰好一个原始数据值。映射表中的映射ID的数值优选地被选择为使得映射ID的数字顺序与原始数据值的顺序关系相同。原始数据值的顺序关系特别是字母顺序、数字顺序或以其他方式定义的顺序。可搜索阵列中元素的排序对应于基于映射ID的数字顺序的搜索顺序。

[0112] 这可能是有利的,因为否定列表和/或肯定列表的评估可以非常有效地执行。例如,对于B树,不需要在树的节点中进行顺序搜索,因为树中的节点是根据顺序关系排列的。当考虑到元素的数量少,构建搜索树的成本太高时,使用排序列表可能特别有利。根据本发明实施例,还可以生成搜索树和搜索列表的组合,其中,阵列是作为搜索树还是搜索列表生成的问题是根据要搜索的数据值的数量而动态生成的,其中,只有当要搜索的数据值的数量超过最小数量时才生成搜索树。

[0113] 根据实施例,字段特定数据值列表各自都是非冗余数据值列表,这些非冗余数据值列表选择向在逻辑数据记录中表示字段特定数据值列表的字段分配的那些数据值。各个字段特定数据值列表中的每个数据值都是唯一的(因此是“非冗余”列表),并且被存储为链接到由字段特定数据值列表表示的字段中包含该数据值的所有逻辑数据记录的数据记录ID。

[0114] 数据值优选地以有序形式存储在字段特定数据值列表中。这可能会极大地提高这些列表中数据库查询的搜索速度。

[0115] 本发明实施例可以具有这样的优点,即由与许多不同字段(表示特定属性或语义概念)相关的大量(甚至每个对象数千个)数据值描述的数据记录可以用非常短的查询时间来查询最多样字段特定数据值的任何组合,而不必遵守依赖于数据对象的原始结构的规范(例如,在关系型、基于索引的DBMS中关于数据库中的表和索引结构的搜索查询的情况)。在基于索引的系统中,所有键的所有可能组合的索引都必须可用。因此,传统的基于索引的DBMS中的索引量会随着键的阶乘而增长!特别是在具有大量不同语义概念的大量不同对象类型的情况下,所需索引的数量通过与键相关的搜索标准与键的阶乘的任何可想象的组合而增长!另一方面,根据本发明实施例,索引的生成和使用(在除了实际数据值之外生成的可搜索数据结构的意义上)是不必要的,特别是在数据库的静态部分中是不必要的。根据实

施例,非冗余数据值列表对应于每个字段,其中该字段的每个数据值一与例如关系DBMS的基于数据记录的表相反一仅出现一次。因此,根据本发明实施例,可以执行搜索和/或分析,而不必为期望的搜索查询生成合适的索引结构。相反,可以直接在无冗余字段特定数据值列表中执行搜索。

[0116] 根据本发明实施例,合并包括接收对最近合并时间点以来指示的更改进行合并的命令。该命令在第二时间点处被接收。响应于接收到命令,方法包括:

[0117] -实现在最近合并时间点和第二时间点之间指示的字段特定数据值列表或其副本中的更改,以便生成合并字段特定数据值列表,使得即使在考虑了在第一时间点和第二时间点之间指示的该字段中的更改之后,合并字段特定数据值列表之一的每个数据值仅被分配包含该数据值的那些逻辑数据记录的ID。

[0118] -在第二时间点之后使用合并字段特定数据值列表而不是先前使用的字段特定数据值列表来执行数据库搜索;

[0119] -使用第二时间点作为新的最近合并时间点。

[0120] 例如,可以通过从对应的字段特定列表中移除包含在否定列表中的数据记录ID来合并字段特定数据值列表,其中,数据记录ID仍然链接到对应的数据值。类似地,合并可以包括将肯定列表中的数据记录ID添加到相应的字段特定列表,使得这些数据记录的ID现在也链接到相应的数据值。

[0121] 根据实施例,响应于接收到命令并且在生成合并字段特定数据值列表之后,该方法包括重新生成至少一个数据结构。该生成基于合并字段特定数据值列表,其中,数据结构的重新生成包括清空肯定列表和/或否定列表。

[0122] 这可能是有利的,因为在合并过程中,肯定列表和否定列表的内容在字段特定数据库列表中被转换,从而在第二时间点处在数据库中创建数据的新的合并状态。对于在第二时间点之后并且因此在合并之后接收到的执行数据值更改的指令,清空的肯定列表和否定列表被再次填充,并且这继续直到再次执行合并为止。这防止肯定列表和否定列表变得太长,这将使它们的使用和管理(例如生成和更新B树)在计算上更加昂贵。

[0123] 根据本发明实施例,实现在最近合并时间点和第二合并时间点之间指示的更改包括通过分析受更改影响的所有字段特定数据值列表中的所有数据值的肯定列表和否定列表,来确定在最近合并时间点和第二合并时间点之间所指示的更改。

[0124] 根据本发明实施例,逻辑数据记录中的至少一些包括一个或更多个“上级”字段,其中,“上级”字段中的每一个被配置为存储从属于该数据记录的数据记录的数据记录ID。这意味着DMS系统被配置为仅将数据记录ID存储在实际满足所述关系的数据记录的“下级”字段中。

[0125] 字段特定数据值列表包括表示“下级”字段的数据值列表,其中,存储在该数据值列表中的数据值是从属于至少一个其他逻辑数据记录的逻辑数据记录的ID。“上级”数据值列表中的数据值中的每一个都被分配其他下级数据记录的一个或更多个ID。数据库查询包含完整性搜索参数,完整性搜索参数指定除了在数据库查询中确定的数据记录之外,是否还要输出从属于这些数据记录的数据记录。

[0126] 数据库查询的执行包括:

[0127] -确定下级数据记录也要被输出;

[0128] -用在步骤i)中确定的数据记录ID搜索“上级”数据值列表,以获得从属于在步骤i)中确定的数据记录的数据记录的一个或更多个ID;

[0129] -评估地址分配表,以识别向所确定的下级数据记录之一的ID分配的AOD条目的地址;以及

[0130] -访问AOD条目的这些识别出的地址,以便将下级数据记录添加到数据库查询中确定的数据记录中。

[0131] 这可能是有利的,因为还可以输出层次对象关系,其中,从步骤i)中的数据记录层次结构中的搜索值的查询结果开始,层次结构向上扩展一个或更多个级别。如果在步骤i)中确定其ID的数据记录从属于若干其他数据记录,例如因为它是具有若干拥有者的车辆,则可以同时输出上级数据记录及其数据值,而不必对若干表执行复杂且低效的数据库查询。

[0132] 根据本发明实施例,逻辑数据记录中的至少一些包括一个或更多个“上级”字段,其中,“上级”字段中的每一个被配置为存储从属于该数据记录的数据记录的数据记录ID。

[0133] 字段特定数据值列表包括表示“下级”字段的数据值列表。存储在该数据值列表中的数据值从属于至少一个其他逻辑数据记录的逻辑数据记录的ID,其中,“上级”数据值列表中的每一个被分配其他下级数据记录的一个或更多个ID。

[0134] 数据库查询包含完整性搜索参数,完整性搜索参数指定除了在数据库查询中确定的数据记录之外,是否还要输出从属于这些数据记录的数据记录。数据库查询的执行包括:

[0135] -确定下级数据记录也将被输出;

[0136] -用在步骤i)中确定的数据记录ID搜索“上级”数据值列表,以获得从属于在步骤i)中确定的数据记录的数据记录的一个或更多个ID;

[0137] -评估地址分配表,以识别向所确定的下级数据记录之一的ID分配的AOD条目的地址,

[0138] -访问AOD条目的这些识别出的地址,以便将下级数据记录添加到数据库查询中确定的数据记录中。

[0139] 这可能是有利的,因为还可以输出层次对象关系,其中,从步骤i)中的数据记录层次结构中的搜索值的查询结果开始,层次结构向下扩展一个或更多个级别。如果在步骤i)中确定其ID的数据记录是若干其他数据记录的上级,例如因为该数据记录表示拥有若干车辆的人,其中,车辆被表示为从属于该人的数据记录,则可以同时输出下级数据记录及其数据值,而不必对若干个表执行复杂且低效的数据库查询。

[0140] 因此,实施例的优点在于,数据库查询的结果可以动态扩展,使得除了直接确定为(在步骤i)中的)数据库查询的结果的数据记录之外,还可以确定并输出在这些初始结果数据记录之上或之下的其他数据记录。这里不需要对数据库查询进行调整。这意味着,只返回在步骤i)中确定的数据记录(富含数据值)的数据库查询和除了结果数据记录之外还包含下级和/或上级数据记录(连同它们的数据值)在内的数据库查询可以以几乎相同的速度执行。为此,既不需要调整查询并使其更复杂(从而降低性能),也不需要为不同的查询选项预先定义单独的SQL SELECT查询(没有补充数据记录、只补充上级数据记录、仅补充下级数据记录、补充下级和上级数据记录)。由于上级和下级数据记录的确定最终基于经由集合运算的数据记录ID,并且经由地址分配表和一个或更多个AOD条目以高性能的方式执行这些数

据记录ID与数据值(属性)的补充,根据实施例的数据库是高度灵活的、高度可扩展的,甚至可以在CPU和存储器需求最小的情况下对非常大的数据记录执行最复杂的查询。

[0141] 根据实施例,上面描述的用于用另外的下级和/或上级数据记录来补充在步骤i)中最初确定的结果数据记录的步骤也可以迭代地执行,使得还可以输出几个级别的上级和下级数据记录。通过这种方式,可以高性能地处理最高复杂度的数据库查询,这在传统的DBMS中是不可能的,至少在可接受的运行时间内是不可能。

[0142] 根据实施例,DMS系统具有接口,该接口允许查询系统(例如软件、另一计算机系统、用户的数字表示等)不仅指定数据库查询并使其执行,而且指定数据库查询除了实际结果数据记录之外是否应确定并输出,以及该查询的实际结果数据记录的上级和/或下级数据记录。查询系统可以优选地针对每个单独的查询单独地指定该查询是否要作为这样的扩展查询来执行,并且DMS系统通过在“上级”和/或“下级”字段特定数据值列表中执行搜索来自动检查对于在步骤i)中为查询确定的数据记录是否存在上级或下级数据记录。

[0143] 根据一些实施例,每个AOD条目都包含标识符,也被称为“标志”,该标识符指示AOD条目所涉及的数据记录是否存在上级或下级数据记录。如果该标识符指示不存在上级或下级数据记录,则即使查询实际上是作为扩展查询执行的,也会省略搜索“上级”和/或“下级”数据值列表的步骤,因为在这种情况下,该标识符已经指示不存在该数据记录的上级或下级数据记录。这可以进一步提高该方法的速度。

[0144] 根据本发明实施例,字段特定数据值列表包括多个被称为时间值列表的字段特定数据值列表。每个时间值列表由时间点的非冗余列表组成,其中,时间点各自表示该时间值列表所涉及的字段的数据值的有效性在一个或多个逻辑数据记录中开始或结束的时间点。时间值列表中的每个时间点被分配在该时间点处有效的逻辑数据记录的ID。例如,“名字”字段的字段特定数据列表可以包含所有数据记录的所有名字的非冗余列表,并且与该字段相关联的时间值列表可以包含名字字段的数据值在任何数据记录中变为有效(例如,通过生成具有名字的数据记录或通过当前有效的名字分配给现有数据记录的名字字段)的所有时间点处的非冗余列表。

[0145] 该方法优选地进一步包括:

[0146] -响应于接收到与逻辑数据记录中的一个相关(即与将新的、更当前的数据值分配给数据记录的字段之一相关)的更改请求,生成要更改的数据记录的新版本,其中,新版本是包含至少一个预版本字段的新逻辑数据记录,其中,预版本字段包含要更改的数据记录的ID,其中,新版本而不是要更改的数据记录包含更改请求中指定的更改和更改时间;以及

[0147] -将具有新数据记录ID的数据记录的新版本存储在字段特定数据值列表中,并存储时间值列表的新数据记录的有效性的开始以及所述更改指令所涉及的字段,其中,新版本的ID存储在时间值列表中。

[0148] 这可能是有利的,因为数据记录的有效性及其单个数据值的有效性的历史过程也被存储,使得数据记录随时间的历史也被存储并且可以被重建。如果数据记录的数据值更改,即先前的数据值变为无效,则DMS系统创建新的逻辑数据记录,其有效性的开始始于新数据值的有效性或分配的开始,其中,当数据记录的数据值之一变得无效(即更改或删除)时,数据记录的有效期结束。

[0149] 根据本发明实施例,数据库查询包括字段特定有效性时间的指示。有效性时间指

定在数据库查询中要确定的数据记录版本在对应字段中包含至少一个字段特定搜索值的时间点或时间段。数据库查询的执行包括：

[0150] -识别与搜索值和有效性时间相关的字段所关联的时间值列表；

[0151] -用有效性时间搜索识别出的时间值列表,以便识别存储在与有效性时间或有效性时间段内的时间点相链接的时间值列表中的那些数据记录版本的一个或更多个数据记录ID;

[0152] -识别与搜索值和有效性时间相关的字段所关联的字段特定数据值列表；

[0153] -用搜索值搜索识别出的数据值列表,以便识别存储在与搜索值相链接的数据值列表中的一个或更多个数据记录ID;

[0154] -计算基于识别出的时间值列表和识别出的数据值列表确定的数据记录的版本的ID的交集；

[0155] -评估地址分配表,以识别向在先前步骤中确定的数据记录版本之一的ID分配的AOD条目的地址;以及

[0156] -访问在先前步骤中识别的AOD条目的地址,以便通过它们的字段值来补充在有效性时间点处或在有效性时间段期间有效的数据记录版本并输出该数据记录版本。

[0157] 这可能是有利的,因为对于数据记录的每个字段值,可以确定它何时被更改或者在什么时间段期间数据值对于特定数据记录的特定字段是有效的。由于时间值作为数据值存储在非冗余数据值列表中,因此也可以有效地搜索在某个时间点上、之后或之前发生的更改。时间信息可以例如由日期和时间信息的组合组成,和/或可以存储为从预定义全局开始时间开始随时间增加的数值。

[0158] 根据本发明实施例,字段标识符的数量和/或类型对于至少一些数据记录是不同的。

[0159] 这可能是有利的,因为它使数据库的使用非常灵活。可以在数据库中存储各种各样的数据,而不必调整物理存储结构的结构,在这种情况下是字段特定数据值列表。在数据库中存储原始数据时,可以避免信息丢失,因为数据被转换为狭窄的、预定义的表结构。相反,具有不同字段的数据记录可以平等地存储在数据库中,因为最终每个字段的数据值存储在非冗余字段特定数据值列表中。

[0160] 根据一些实施例,逻辑数据记录还可以包括引用同一字段的多个字段标识符-数据值对。这对于表示关系的字段来说可能是特别有利的。例如“上级”字段可能在数据记录中多次出现,该数据记录表征了拥有多辆车的人。通过将表示该人拥有的车辆的数据记录的数据记录ID存储在该人的多个“上级”字段中,可以表示该人拥有车辆的关系并将其存储在数据库中。通过在数据记录中为同一字段设置多个字段标识符-数据值对,可以存储对象之间的任意关系,包括1:n和n:m关系,以便数据库查询可以非常快速地确定、输出和/或考虑这些复杂关系。

[0161] 根据本发明实施例,AOD条目被存储为仅添加数据结构中的区块链的元素,这些元素经由密码散列值链接在一起。数据库搜索的执行包括对在数据库查询过程中处理的那些AOD条目的散列值的有效性检查。

[0162] 这可能是有利的,因为可以有效地防止对仅添加数据结构的后续操纵,或者至少可以通过DMS系统在使用仅添加数据结构之前验证区块链的散列值来立即检测这种操纵。

[0163] 在另一方面,本发明涉及一种易失性或非易失性存储介质,计算机可读指令存储在该介质上。指令被配置为使得处理器执行根据本文所述的任何实施例和示例的用于在数据库内执行数据库查询的方法。

[0164] 在另一方面,本发明涉及一种用于在数据库内执行数据库查询的计算机实现的方法,并且涉及相应的计算机系统、计算机程序产品和数据结构。数据库在第一时间点(被称为“最近合并时间点”)处包含多个逻辑数据记录。每个逻辑数据记录包含数据记录ID和一个或多个字段标识符-数据值对。数据记录以字段特定数据值列表的形式物理地存储。该方法包括在最近合并时间点之后:

[0165] -接收更改数据记录中若干数据记录的字段的数据值的指令;

[0166] -在不更改字段特定数据值列表的情况下,将指令存储在仅添加数据结构中,其中,仅添加数据结构中的每个条目(此处被称为AOD条目)至少包含根据更改指令之一要更改的数据记录之一的字段标识符-数据值对中的那些条目;

[0167] -对于以下数据记录中的每一个:即关于该数据记录在最近合并时间点之后数据库接收到更改数据值的一个或多个指令,在地址分配表中存储对数据记录的更改进行指定的被存储AOD条目中的最近AOD条目的地址,数据记录链接到该数据记录的数据记录ID,其中,地址分配中的链接被自动更新;以及

[0168] 使用地址分配表对字段特定数据值列表执行数据库查询。例如,数据库查询可以包括上面描述的步骤i-步骤iv,这里针对本发明实施例进行了描述。所有其他实施例也可以与本发明的这一方面相结合。在另一方面,本发明涉及数据结构。该数据结构包括:

[0169] -多个字段特定数据值列表,其中以分布式方式存储逻辑数据记录,其中,每个数据记录包含数据记录ID和一个或多个字段标识符-数据值对;

[0170] -仅添加数据结构,包含用于更改数据记录的字段的数据值的指令,其中,仅添加数据结构中的每个条目(此处被称为AOD条目)至少包含根据更改指令之一而要更改的数据记录之一的字段标识符-数据值对中的那些条目;

[0171] -地址分配表,其中,地址分配表向以下数据记录中的每一个的ID分配对该数据记录的更改进行指定的最近AOD题目的地址:针对该数据记录,具有更改指令的AOD条目被存储在仅添加数据结构中。

[0172] 每个字段特定数据值列表、仅添加数据结构和地址分配表各自都被视为一个数据结构。

[0173] 根据本发明实施例,数据结构附加地或替代地包括至少一个数据结构,该数据结构包含可搜索的、有序的元素阵列。阵列是列表元素的列表或节点的搜索树。特别地,搜索树可以是B树。阵列在每种情况下表示逻辑数据记录的字段之一。阵列的元素表示包含在数据记录中并分配给由阵列表示的字段的数据值的非冗余列表中的一个数据值。阵列的元素中的每一个被存储为与空的或非空的肯定列表和/或空的或非空的否定列表相链接。

[0174] 根据本发明实施例,数据结构附加地或替代地包括字段特定无冗余数据值列表。数据值列表可以被配置为从原始数据获得的字段特定原始数据值列表,原始数据记录已经存储在字段特定原始数据值列表中,或者在映射表中唯一分配给这些原始数据值的值(优选为数值)列表(这里被称为“映射ID”)。

[0175] 根据本发明实施例,数据结构附加地或替代地包括映射表。映射表向原始数据记

录的每个原始数据值分配未分配给任何其他原始数据值的至少一个映射ID。映射ID用作逻辑数据记录和字段特定列表的数据值。

[0176] 根据本发明实施例,数据结构附加地或替代地包括一个或多个字段特定时间值列表,其中,每个时间值列表是由有效性时间的无冗余列表组成的列表,其中,每个有效性时间指示表示该时间值列表的字段的数据值在逻辑数据记录之一中被更改的时间,其中,每个字段特定时间值列表中的每个有效性时间被分配逻辑数据记录的所有版本的数据记录ID,其中与时间值列表的字段相对应的数据值在该有效性时间处被更改。

[0177] 根据本发明实施例,数据结构附加地或替代地包括一个或多个“上级”数据值列表,其中,存储在该“上级”数据值列表中的数据值是从属于至少一个其他逻辑数据记录的逻辑数据记录的ID,其中,“上级”数据值列表中的每个数据值被分配给其他下级数据记录的一个或多个ID。

[0178] 根据本发明实施例,数据结构附加地或替代地包括一个或多个“下级”数据值列表,其中,存储在该数据值列表中的数据值是从属于至少一个其他逻辑数据记录的逻辑数据记录ID,其中,“下级”数据值列表中的每个数据值被分配给其他上级数据记录的一个或多个ID。

[0179] 在另一方面,本发明涉及一种计算机系统,该计算机系统至少包括处理器和包括数据库的数据存储器。

[0180] 数据库在被称为“最近合并时间点”的第一时间点处包含多个逻辑数据记录,其中,每个数据记录具有数据记录ID和一个或多个字段标识符-数据值对。每个数据记录包含数据记录ID和一个或多个字段标识符-数据值对,其中,数据记录以字段特定数据值列表的形式物理地存储。

[0181] 该计算机系统还包括数据管理和搜索系统—DMS系统。DMS系统被配置为管理数据库,其中,在最近合并时间之后的管理包括:

[0182] -接收更改数据记录中的若干数据记录的字段的数据值的指令;

[0183] -在不对字段特定数据值列表进行更改的情况下,将指令存储在仅添加数据结构中,其中,仅添加数据结构中的每个条目(此处被称为AOD条目)至少包含根据更改指令之一而要更改的数据记录之一的字段标识符-数据值对中的那些条目;

[0184] -对于以下数据记录中的每一个:即关于该数据记录数据库在最近合并时间点之后接收到更改数据值的一个或多个指令,在地址分配表中存储对该数据记录的更改进行指定的被存储AOD条目中的最近AOD条目的地址,该数据记录链接到该数据记录的数据记录ID,其中,地址分配中的链接被自动更新;以及

[0185] -执行数据库查询,其中,数据库查询包括:

[0186] i. 基于与一个或多个字段特定搜索值的匹配,搜索字段特定数据值列表以识别内容要被全部或部分返回的数据记录(214)的ID;

[0187] ii. 评估地址分配表,以便识别分配给i)中识别出的数据记录ID之一的AOD条目的地址,

[0188] iii. 访问识别出的AOD条目的地址;以及

[0189] iv. 使用包含在这些识别出的AOD条目中的更改细节,以将字段标识符-数据值对添加到在步骤i)中确定的数据记录ID,并输出后者。

[0190] 根据实施例,至少一个处理器包括ALU(算术逻辑单元处理器)或FPGA(现场可编程门阵列)。ALU或FPGA被配置为对数据记录ID的集合执行集合运算,其中,集合运算特别包括交集、并集、差集或对称差集的计算。特别地,可以由ALU或FPGA执行集合运算,使得在ALU或FPGA的单个运算周期(比较运算)内执行两个数据记录ID的比较。例如,标识符的长度可以对应于处理器架构的处理范围(例如32位用于32位架构、64位用于64位架构)。如果处理器架构能够特别有效地处理数值,则标识符可以由数值组成。如果处理器架构能够特别有效地处理其他值类型(例如符号),则标识符可以由符号组成。

[0191] 这可能是有利的,因为其设计(例如寄存器大小)被优化为对预定义长度的数据记录ID执行集合运算的硬件部件形式的实现可以显著加快数据库中数据的处理和搜索。

[0192] “处理器”在这里被理解为一种(通常尺寸上大大缩小并且通常空闲的)可编程算术单元(即机器或电子电路),它根据传输的命令控制其他机器或电路,并在这样做的过程中驱动算法(过程),该算法通常涉及数据处理。例如,处理器可以是主处理器、中央处理单元,或者(更一般地)用于计算机或在其中执行命令的计算机类设备的中央处理单元(简称CPU)。处理器也可以被配置为嵌入式系统(例如家用电器、售票机、智能手机等)中的微控制器。

[0193] 根据实施例,该方法的至少一个步骤由处理器的子单元直接执行。

[0194] 特别地,根据本发明实施例,对数据记录ID的集合运算可以由至少一个处理器的算术逻辑单元(通常缩写为ALU)直接执行。ALU可以将具有相同位数(n)的两个二进制值链接起来。为n位ALU做出参考。n的典型值为8、16、32和64。根据本发明实施例,所有数据记录ID都具有固定的长度,特别是相同的长度,其优选地被选择为使得每个数据记录ID完全适合ALU的工作寄存器,并且可以在ALU中的集合运算的过程中相互比较。特别地,为了比较两组数据记录ID,例如,为了计算交集、并集、差集或对称差集,ALU可以将一个集合的所有数据记录ID与另一集合的全部数据记录ID进行身份比较。

[0195] 这里,“原始数据”是指以电子形式提供但尚未以DMS系统的解析器解析的形式提供的任何数据。特别地,原始数据包括在观察、测量或数据收集期间直接获得的并且未经处理仍然可用的数据。

[0196] “数据处理和搜索系统-DMS系统”在这里被理解为用于存储、管理和处理电子数据的基于软件和/或硬件的系统。根据本发明实施例,DMS系统被配置为有效地、无矛盾地并且永久地存储大量数据。根据实施例,DMS系统可以包括几个部件,这些部件可以是模块的形式:用于接收和解析原始数据以及用于将解析的数据存储在非冗余列表中的导入部件。在这里,导入部件可以使用现有列表,并且在必要时可以自动创建新列表。DMS系统可以进一步包括用于搜索和/或分析列表的搜索和分析部件。可选地,DMS系统可以包括图形用户界面(Graphical User Interface,GUI),该GUI允许用户配置搜索,以便例如指定是否也要输出下级和/或上级数据记录或某些数据记录的历史。DMS系统具有对由DMS系统管理的数据存储器的读写访问。根据实施例,DMS系统包括数据存储器 and 存储在其中的数据值列表。可选地,DMS系统可以进一步包括用于存储至少一些原始数据的文档存储器。

[0197] “逻辑数据记录”在这里被理解为具有相关内容(例如项目编号和项目名称和生产日期)的一组数据值(属于对象)。数据记录对应于可识别的数据值的逻辑结构,例如,在解析原始数据或导入基于字段的原始数据时。数据记录的逻辑结构指定此数据记录包含多少

字段以及哪些字段,以及将哪些数据值分配给这些字段中的每个字段。然而,在物理上,数据记录可以不同地存储,特别是以分布式方式存储,例如数据记录的数据值可以各自存储为无冗余字段特定数据值列表的元素。

[0198] 包含在原始数据中的数据对象可能已经包含数据记录ID(例如具有面向行的数据记录的Excel表的行号),但是根据实施例,数据记录ID也可能由DMS系统在解析或导入原始数据的过程中动态创建。特别地,可以生成数据记录ID以显式地或隐式地对分配给该数据记录的地址分配表中的行的存储器地址进行编码。

[0199] 逻辑数据记录可以是原始数据记录,但优选地,它是在转换步骤中通过用映射ID替换原始数据值而从原始数据记录形成的数据记录。

[0200] “原始数据记录”在这里被理解为指一组具有相关内容的的数据值(属于对象),在这里被称为“原始数据值”,“原始数据值”是从原始数据中获得的、自动导入的或由用户输入的。原始数据记录可以至少部分由非数字的数据值(例如自然语言的词语)组成。

[0201] “仅添加数据结构”在这里被理解为指一种数据结构,尤其是文件,在该数据结构中,新数据可以添加到数据结构的一端,但现有数据是不可更改的。许多文件系统的访问控制列表实现了“仅添加”权限。例如,带有`chattr`的Linux操作系统支持在文件和目录上设置“仅添加”标志。一些云存储提供商将限制访问的选项提供为“仅添加”。到目前为止,该功能已被用于降低备份策略的数据丢失风险,但并不是为了与地址分配表相结合来加快数据库查询,因为使用这些数据记录的字段相关数据值来丰富结果数据记录ID基于这些数据结构执行。仅添加数据结构会随着时间的推移而生长。

[0202] “地址”在这里是指逻辑或物理存储器区域的唯一标识符。优选地,存储器区域是可在地址的帮助下直接访问的易失性或非易失性存储介质(例如RAM、硬盘驱动器等)内的区域。该地址在存储器访问中用于指定进行访问的确切位置。寻址的细节取决于特定硬件。AOD条目的地址可以是复合地址,其第一部分指定第一写入的AOD条目的存储器地址,其第二部分指定相对于第一AOD条目的偏移量,在该偏移量处可以找到由复合地址指定的AOD条目。该地址可以是逻辑地址或物理地址。

[0203] 特别地,“地址”可以被定义为使数据处理系统能够至少对在该地址处可用的数据进行读取访问的指示。例如,地址可以是经由网络可用的文件的URL、文件的基于本地文件系统的地址、或者特定文件内的条目等。

[0204] “数据值”在这里被理解为指数据记录中最小的可评估单元。例如,数据值可以是字符串、图像、像素矩阵、Unicode字符或数值。例如,可以在数据导入、解析步骤和/或标记化步骤的过程中,或者在这些步骤之后的映射步骤的过程中获得数据值,在这些步骤中,原始数据值被分配数字映射ID,数字映射ID最终用作数据值,被分配给字段并存储在字段特定数据值列表中。

[0205] “无冗余数据值列表”在这里被理解为每个数据值只包含一次的数据值列表。优选地,数据值按排序顺序存储在列表中,以加快对列表的数据库查询。

[0206] “字段特定数据值列表”在这里被理解为指分配给特定字段并表示该字段的数据值列表,该列表选择性地仅包含确切地该字段中的数据记录中包含的数据值。数据值也可以是分配给其他(“原始”)数据值的映射ID。

[0207] “时间值列表”在这里被理解为指无冗余数据值列表,其数据值是时间点或时间点

的映射ID。时间值列表优选地在这里被分配给数据记录的字段之一。存储在时间值列表中的每个时间值表示时间值列表所分配到的字段的数据值在一个或多个数据记录中发生了更改的时间点。受该时间点处的更改影响的数据记录的ID链接到时间值列表中的该时间值进行存储。

[0208] 由于数据值的内容的更改通常也存储在字段特定数据值列表中(例如以分别分配给先前的和新的字段相关数据值的数据记录ID的更改的形式),该字段特定数据值列表和字段特定时间值列表的内容的组合揭示了数据值更改的内容和时间两者。

[0209] 有效期在这里被理解为指某个数据值被分配给数据记录的字段(因此对该数据记录和该字段“有效”)的一段时间。例如,通过所述数据值被分配给数据记录的字段的第一有效性时间(例如通过第一次生成或导入数据记录或通过重写废弃数据值),以及通过该数据值被重写或数据记录被删除或被视为废弃版本的有效性时间段结束,该时间段受到限制。

[0210] “有效性时间点”在这里被理解为指某个数据值被分配给数据记录的字段(因此对该数据记录和该字段“有效”)的时间点。

[0211] “数据存储器”是指用于存储数据的存储介质或存储介质上的存储区域或几个存储介质或存储器区域的组合。如果数据存储器包括多个存储介质或存储介质区域,则这些存储介质或存储介质区域可以相互连接以形成逻辑数据存储器。在这种情况下,存储介质或存储介质区域可以例如经由网络或经由计算机系统的总线可操作地彼此连接。例如,由DMS系统管理的数据存储器可以是DMS系统对其数据具有独占访问权的数据存储器。

[0212] “字段”在这里被理解为表示特定含义内容的参数(或“属性标识符”),并且可以在各种特定实施例实现。例如,字段可以表示对象的特性,例如颜色、建筑类型、宽度、高度、年龄、密度等。根据存储在数据记录中的数据的内容,字段可能不同。例如,表示机器的数据记录通常与表示人的数据记录具有不同的字段。根据所使用的原始数据和/或解析器的类型,解析和导入的原始数据值和/或分配给它们的映射ID可能具有分配给它们的各种不同的字段。例如,对于医学数据,一个字段可以表示“诊断”的概念,其可以被分配各种特定实施例(“数据值”或“字段值”),例如“糖尿病”、“帕金森病”或“皮肤癌”。另一个字段可能表示“症状”,将“发烧”、“发冷”、“牙痛”(或它们各自的映射ID)等实施例或数据值分配给该字段。数据值也可以分配给不同的字段。例如,数据值“银”可以分配给字段或概念“金属”,也可以分配给概念“颜色”或“姓氏”。根据实施例,分配给几个不同字段的数据值的映射ID可以根据字段而不同或相同。

[0213] “计算机系统”在这里被理解为指单片或分布式数据处理系统,特别是数字数据处理系统。因此,数据处理系统可以由例如独立的计算机系统或计算机网络(特别是云系统)组成。计算机系统还可以被配置为例如移动数据处理系统,例如笔记本电脑、平板电脑或便携式电信设备(例如智能手机)。

[0214] “系统”在这里被理解为指能够处理数据的一个或多个元素(例如计算机系统)的总和。为此目的,系统部件交换数据和/或控制命令。例如,系统可以包括具有DMS系统的计算机。可选地,该系统可以包括向DMS系统发送搜索请求和/或分析请求的另外部件,例如一个或多个客户端计算机系统。

[0215] “数据结构”在这里被理解为指用于存储和组织数据的对象。特别是,它是一种结构,因为数据是以某种方式排列和链接的,以使它们能够被有效地访问和管理。

[0216] “合并时间点”在这里被理解为时间点。特别地,合并时间点可以是DMS系统的特定数据与DMS系统的其他数据集成的时间点。例如,AOD数据结构、否定列表和/或肯定列表中的数据包含对数据值的更改和/或关于将数据值分配给逻辑数据记录的更改,数据可以在合并过程中被集成到字段特定数据值列表中,使得字段特定列表在合并之后也包含所述更改。

附图说明

- [0217] 在下文中,参照附图描述本发明的实施例。在附图中
- [0218] 图1示出包括DMS系统的系统的框图;
- [0219] 图2示出用于指定搜索查询的GUI的框图;
- [0220] 图3示出数据库的不同数据结构的框图;
- [0221] 图4示出不同类型的原始数据的示例;
- [0222] 图5示出将原始数据转换为字段特定数据值列表的示例;
- [0223] 图6示出用于使用数据库的合并版本和非合并版本的分布式系统的框图;
- [0224] 图7示出用于执行数据库查询的方法的流程图;
- [0225] 图8示出用于考虑数据库查询中尚未合并的数据值变化的方法的流程图;
- [0226] 图9示出具有多个肯定列表和否定列表的字段特定数据结构的示例;
- [0227] 图10示出合并之前和之后的数据结构的框图;以及
- [0228] 图11示出有效输出完整数据记录的数据结构。

具体实施方式

[0229] 图1示出具有DMS系统102的用于对数据库104执行数据库查询的系统100的框图。系统100尤其可以是一个或更多个处理器的计算机系统,例如数据库服务器。

[0230] 例如,该系统可用于执行图7和图8中所示的方法。因此,在下文中,系统100和图7的方法将参考两个相应的图一起进行描述。

[0231] 例如,该系统和方法可以用于集成大量异构结构的原始数据112,并使其以有效和灵活扩展的方式可搜索和分析。

[0232] 例如,原始数据可以包括不同内容和结构的XML文件、JSON文件、各种格式的文本文件(例如Microsoft Word、OpenOffice、Adobe Acrobat PDF文件、TXT文件等)、来自一个或更多个不同关系数据库的不同表、媒体数据或分层组织的数据(例如对象树)。在一些原始数据中,数据对象、它们的数据值以及可选地还有它们的语义含义可以或多或少地被明确地标识,例如在数据库表、Excel文件和其他具有相应字段和字段标识符的相对高度结构化的数据中。在其他原始数据(例如图像数据)中,数据对象及其数据值可能不是显式的,而是隐式的。这意味着数据对象及其数据值仅在解析过程中被识别和提取。

[0233] 可以使用几个不同的句法和/或语义解析器118-130来解析各种原始数据。解析器优选地在解析期间与DMS系统或DMS系统的其他部件进行交换,例如当动态地创建与逻辑数据记录相对应的数据对象的数据记录ID时,能够确定该ID是真正唯一的还是已经被数据值列表之一中使用的数据记录ID占用。

[0234] 数据记录ID可以是已经由原始数据预定义的逻辑数据记录的标识符,并且被DMS

系统采用作为数据记录ID。替代地,数据记录ID也可以在数据记录的导入或生成期间生成。

[0235] 生成的、解析的和/或直接导入的逻辑数据记录的数据值根据由原始数据的字段标识符或由解析器分配给它们的语义概念存储在相应的字段特定数据值列表中。

[0236] 在几个字段特定数据值列表中存储逻辑数据记录在一定程度上解决了对象结构,即特定数据对象中存在哪些数据值或语义概念或者存储器上数据值的物理组织不面向它们与数据记录的逻辑隶属关系的问题。

[0237] 在逻辑数据记录已成功存储在数据库104中之后,数据库最初在“第一时间点”处处于一致状态。第一时间点也被称为“最近合并时间点”。在这个时间点处,数据库因此包含先前以字段特定数据值列表的形式存储在数据库中的全部逻辑数据记录,包括映射表。字段特定数据值列表的整体表示在第一合并时间点处的数据库的静态部分101。

[0238] 在稍后的时间602处,DMS系统102接收用于更改数据值的各种指令,例如用于删除或更改单个数据值或用于删除或添加整个逻辑数据记录的指令。例如,可以从作为软件程序或硬件部件或者是自然人的数字表示的查询系统接收指令。

[0239] 这些接收到的更改指令最初不对字段特定数据值列表执行。相反,指令被存储604在仅添加数据结构202中,而不执行对字段特定数据值列表116的更改。仅添加数据结构中的每个条目(此处被称为AOD条目)至少包括根据更改指令之一要更改的数据记录之一的字段标识符-数据值对的那些条目。可选地,DMS系统可以被配置为创建和存储AOD条目,使得它们不仅包含更改的数据值,而且包含AOD条目所涉及的数据记录的所有当前数据值(所谓的“完整”或“完全加载”AOD条目)。例如,DMS系统可以被配置为在接收与特定数据记录相关的特定最小数量的更改的请求时将AOD条目存储为完整AOD条目。用于存储完整AOD条目的另一个标准可以是超过搜索查询持续时间。因此,通过引用AOD条目的数量

[0240] 对于以下数据记录中的每一个:即关于该数据记录数据库在最近合并时间点之后接收到更改数据值的一个或更多指令,DMS系统102在地址分配表中存储对该数据记录的更改进行指定的被存储AOD条目中的最近AOD条目的地址206、208,该数据记录链接到该数据记录的数据记录ID。这个步骤的细节和AOD条目的性质例如在图3中得到了进一步的说明。地址分配表中的链接会自动更新,以便对于每个数据记录,始终存储包含该数据记录的最近更改的AOD条目之一的地址。因此,仅添加结构还包括尚未与数据库的静态部分101合并的更改和补充。仅添加数据结构、至少其截至最近合并时间点的条目、以及稍后描述的一些其他可选数据结构(例如否定列表和肯定列表),属于数据库104的动态部分103。

[0241] DMS系统然后执行610数据库查询。例如,数据库查询可以在复杂数据分析过程中执行,或者响应于从查询系统接收到搜索查询而执行。

[0242] 数据库查询包含一个或更多数据值(也被称为搜索值),它们引用数据记录的每个字段中的一个。

[0243] 数据库查询包括第一步骤i) 612,针对每个搜索值,用搜索值搜索与该搜索值的字段相对应的字段特定数据值列表,以基于与一个或更多字段特定搜索值的匹配来识别数据记录214的ID,该数据记录214的内容将被全部或部分返回。例如,搜索可以包括搜索值名字=皮特(First name=Peter)和城市=柏林(City=Berlin)。优选地,搜索值是映射ID,并且可以例如通过将由查询系统输入的主搜索值与映射表相匹配来确定,以确定分配给该主搜索值的映射ID。例如,“Peter”可以具有映射ID 2392,“Berlin”可以具有映射ID

68672。因此,搜索将包含例如搜索值名字=2392和城市=68672。在名字数据值列表中搜索2392可以产生“名字查询结果”数据记录ID的集合,在城市数据值列表中搜索68672可以产生“城市查询结果”数据记录ID的集合,并且名字查询结果数据记录ID和城市查询结果数据记录ID的交集可以表示在步骤i)中确定的数据记录ID的集合。例如,在步骤i)中,可以确定这两个数据记录ID 5578和5907,这两个数据记录ID都表示名字为Michael并且居住地址在柏林的人。

[0244] 下一步骤ii) 614是评估地址分配表226,以便识别分配给i)中识别出的数据记录ID之一的AOD条目的地址。例如,DMS系统可以被配置为选择新的逻辑数据记录的ID,使得那些ID对地址分配表中与该逻辑数据记录唯一相关联的行的存储器地址进行编码(无论是显式地(例如作为存储器地址)还是隐式地(例如作为地址分配表中的顺序行号)),通过乘以行存储器大小,给出该行相对于基本地址的存储器地址(例如,文件起始)。地址分配表可以通过从步骤i)中确定的数据记录ID中确定地址分配表中唯一分配给这些ID的行的存储器地址来访问,并且使用这些存储器地址进行直接访问而无需搜索过程。

[0245] 例如,在d)中确定的数据记录ID可以包括个人数据记录的ID,其中,该ID可以是例如“5578”,其中,该值也可以是例如地址分配表中唯一分配给该数据记录的行的存储器地址。

[0246] 因此,为了现在输出个人5578的完整个人数据记录,没有必要以任何方式修改步骤i)中使用的数据库查询。在步骤i)中获得的要返回的数据记录的数据记录ID被用作地址分配表中的键(并且在一些实现变型中作为地址分配表行的存储器地址)就足够了,以首先识别唯一分配给在i)中确定的数据记录的地址分配表行,并且评估这些行。所确定的地址分配表行各自都包含AOD条目的地址,该AOD条目包含对该行唯一分配的数据记录的最近更新。DMS系统只需要直接查找此AOD条目地址,以便至少找到根据该AOD条目已经更改的此人的属性(例如电话号码)。

[0247] 在下一步骤iii)中,DMS系统访问616识别出的AOD条目的地址。如果该AOD条目是完全的(或“完整的”)AOD条目,即包含人的所有当前有效数据值,则完全AOD条目中指定的数据值可用于输出包括所有人属性的人5578的数据记录。如果所述AOD条目不包含所有当前数据值,则其优选地至少包含与同一个人5578相关的下一个较旧AOD条目的地址,并且包含与例如地址、电话号码、工作地点等相关的其他更改。如果该下一个最旧AOD条目是完整AOD条目,可以终止对人5578的仅添加数据结构的搜索。如果不是,则包含在该AOD条目中的地址被跟随到下一个最近的最旧AOD条目,并且如果需要,则重复该步骤,直到逻辑人记录的所有字段都有相应的数据值为止。

[0248] 在步骤iv) 618中,DMS系统使用包含在这些识别出的AOD条目中的更改信息来将字段标识符-数据值对添加到在步骤i)中确定的数据记录ID并输出它们。

[0249] 以这种方式,可以确定其ID在步骤i)中被确定的所有数据记录的字段特定数据值,并且可选地还可以确定这些数据记录的上级或下级数据记录和/或历史版本。

[0250] 搜索查询可以以高效的方式通过关系确定数据记录的任何数据值,甚至确定连接到这些数据记录的其他数据记录的数据值,其中,搜索基本上仅基于数据值列表,并且要输出的结果数据记录的完成基本上基于跳转地址。这使得搜索性能和资源效率更高。数据值列表优选地是排序的、无冗余的数值(特别是映射ID)的列表,使得所有步骤i-步骤iv可以

以高性能的方式执行。

[0251] 根据一些示例,DMS系统可以被配置为以规则或不规则的间隔将数据库的动态部分103与静态部分合并。这意味着高速缓存在动态部分中的数据记录更改现在存储在字段特定数据值列表中。例如,合并可以由合并模块105执行,数据库查询可以由另一模块106执行。例如,模块106可以被配置为经由网络接口从一个或多个客户端计算机接收搜索查询,或者经由GUI直接从本地工作用户接收搜索查询。附加地或替代地,模块106还可以包含基于对多个字段特定数据值列表以及可选地否定列表和/或肯定列表的集合运算的多个复杂分析函数。

[0252] 因此,静态部分表示某个合并时间点处的数据库,并可用于计算对该时间点处可用数据的一致性评估和分析。从这个固定时间点开始的所有更改最初仅在数据库的动态部分103中进行管理,该动态部分103特别包括尚未持久化的仅添加数据结构的部分以及肯定列表和否定列表。因此,仅添加数据结构可以由静态部分和动态部分组成,静态部分包括已经合并的更改指令和AOD条目,即保留在字段特定列表中。动态部分包括在最近合并时间点处尚未合并(保留在数据值列表中)的更改。随着每次合并,合并和非合并AOD条目的边界或划分发生变化,其中,每个边界对应于合并时间点。

[0253] 在任何时刻,数据库的这个动态部件都可以被合并,即集成到静态部件101中,并被新的、空的动态部件替换。以这种方式,先前的动态部件变成静态的,并且可以在后台中与先前的静态部件合并以形成新的静态数据库部件。完成后,可以保存和/或删除先前的静态部件和动态部件。

[0254] 合并模块105可以例如提供API,用户和外部程序经由API在搜索时具有使用例如仅静态部件或两个部件或者在必要时仅动态部件作为搜索的基础的选项。

[0255] 实现变体的巨大优势在于在持续运行期间的可更新性,因为所有部件都可能被冻结,以确保在持续不间断运行期间重新组织,包括可优化性。数据库的静态部件和动态部件的并行操作是可能的,数据库的可更新性也是可能的。

[0256] 图2示出用于指定搜索查询的GUI 154的框图。例如,GUI可以包含几个输入字段,例如用于指定一个或多个搜索值和分配给它们的字段的一个或多个输入字段156,以及另外的字段158、160,其允许用户指定除了实际结果数据记录之外是否要返回的与结果数据记录具有逻辑和/或时间关系的另外的数据记录。例如,除了基于与在步骤i)中在字段156中输入的搜索项的匹配而确定的结果数据记录之外,用户可以指定是否还要输出这些结果数据记录的下级或上级或者表示结果数据记录的先前版本的数据记录。在其他示例中,除了这些字段158、160之外或作为这些字段156、158的替代,GUI还可以包括用于输入时间或时间段的字段,使得用户可以将搜索限制在特定的有效性时间或时间段。

[0257] 在一些示例中,GUI可以包括允许指定迭代(数据记录层次结构)的数量的输入字段159,直到确定和输出上级和/或下级数据记录为止。

[0258] 除了GUI之外或作为GUI的替代,DMS系统还可以提供其他接口,特别是API 152,其他软件程序可以经由其他接口输入这里描述的搜索查询和附加参数(确定是否要输出上级或下级数据记录、是否要输出先前版本和/或搜索是否要被限制在有效性时间段内)。

[0259] 在第一步骤i)中,基于字段特定搜索值和字段特定数据值列表来确定数据记录ID的集合,这也可以被视为中间结果162。在进一步的步骤ii) - 步骤iv)中,确定分配给这些数

据记录ID的当前字段值,并输出由字段特定数据值补充的完整数据记录164作为最终结果。输出的数据记录的数据值尤其可以是映射ID或分配给映射表中的映射ID的原始数据的原始数据值。

[0260] 图3示出数据库104的各种数据结构的框图。

[0261] 根据实施例的用于数据存储的数据结构与其他数据库的不同之处在于,除了用于实际数据存储的数字结构之外,不存在和/或不需要或使用索引。数据管理结构能够实现任何组合中的所有查询、选择和评估,即使有大量数据。

[0262] 导入原始数据时,键值对(键是通常表示参数或属性的字段)会转换为无量纲数据记录ID(RID)。数据记录中出现的所有键/值对都将具有相同的RID。

[0263] 因此,对键/值的任意组合的数据库查询最终产生RID的组合,其中交集集、差集、对称差集和/或并集最终产生结果。

[0264] 数据结构包括仅添加数据结构202,例如仅添加文件。对逻辑数据记录的所有更改都按照DMS系统接收到更改请求的顺序写入该文件,并且可能不再更改。例如,这些更改可以仅由数据记录的单个数据值组成,并且可以包括例如对数据记录补充新的或附加的字段数据值对,但是也可以删除字段数据值对或者用另一个数据值替换一个数据值。更改还可以涉及整个逻辑数据记录,并且例如可以提供整个数据记录的删除或初始存储。

[0265] 仅添加数据结构中的每个条目在框202中被表示为线。例如,每个AOD条目可以包含一个或多个标识符(标志)。标志可以例如是位值的形式,并且包括例如数据记录是否存在上级值或下级值、AOD条目是否是完整AOD条目、数据记录的先前版本是否存在等。

[0266] 虚线206标记第一时间点(最近合并时间点)。该线上方的所有AOD条目222在最近合并时间点处已经被合并;该线下方的所有AOD条目224尚未被合并。箭头204表示数据结构的第一AOD条目。例如,它的地址可以用作基本地址,并且所有其他AOD条目的地址是该基本地址与偏移量的组合。箭头208指示数据结构202的结束,根据接收到的更改请求,进一步的AOD条目被连续地添加到该数据结构202。

[0267] 数据结构可以进一步包括地址分配表226。这将AOD条目的地址分配给数据库104中的每个逻辑数据记录。此AOD条目是仅添加结构中的最近AOD条目,它指定对该一个数据记录的更改。例如,在表226中,数据记录ID(“RID”) = 35的数据记录当前被分配AOD条目,声明该字段F3具有数据值“b”(见箭头209)。然而,在更早的时间处,RID 35的数据记录已经分配了不同的数据值“a”。此较早的更改对应于从顶部开始的第三AOD条目。在这个较早的时间点处,在地址分配表中,RID 35被链接到从顶部AOD条目起的第三个的地址,如虚线箭头207所示。然而,该地址已经被箭头209所引用的AOD条目的地址所取代。然而,此AOD条目209包含甚至在该AOD条目之前写入的最近AOD条目207的地址,该地址指代相同的数据记录RID = 35。因此,DMS系统可以简单地跟随AOD条目内的地址到同一数据记录的下一个最近AOD条目,以获得数据记录35的所有字段的所有当前数据值。

[0268] 在图3和其他图中,人们容易理解的数据值已被用于说明目的。然而,优选地,逻辑数据记录的字段特定数据值是在映射表210中唯一地分配给从原始数据获得和/或最初输入的数据记录的数值。这些数值也被称为映射ID,并且优选地被用作存储在基于字段的列表中并且在搜索期间与搜索值匹配的数据记录。映射表210示意性地示出原始数据值(这里:词语)到映射ID的分配。

[0269] 映射表210例如提供表示要通过映射ID(MID)区分的所有数据值的目的。根据一些实现示例,每个数据值仅经由其MID存储在字段特定列表中,从而为每个FID存储在其上下文中出现的所有MID。数据值可以按字段或按数据类型分类,例如字符、文本、整数、浮点、日期/时间戳、UUID值、二进制值等。例如,对于每个字段或对于每个数据类型,可以对MID进行排序,使得映射表210中的MID以与相关联的原始数据值被排序相同的顺序存储。这使得可以对字段特定数据值列表的内容进行排序,使其MID出现在该列表中,而不必首先加载内容本身。这意味着MID将被分配给名字列表Anton-Bertram-Christoph-Doris-Emil,使得在映射表的排序顺序中,Anton的MID的数值在所有其他名字MID之前,Bertram的数值将排在第二位,依此类推。

[0270] 数据结构还可以包括包含排序和可搜索数据值的字段特定数据结构216,数据值各自存储为链接到否定列表220和肯定列表218。这些还可以用于包括尚未合并和存储在数据值列表116中的更改。

[0271] 例如,在接收到具有字段相关搜索值的搜索查询之后,DMS系统可以首先使用映射表210将搜索值翻译为映射ID。该映射ID然后被用于搜索字段特定数据值列表116,字段特定数据值列表116表示被分配搜索值的字段。可选地,在该步骤i)中,还可以搜索数据值或映射ID的有序阵列216,该有序阵列216包含以否定列表和肯定列表的形式存储的尚未合并的更改,以便仍然以这样的方式扩展和/或减少基于数据值列表116获得的数据记录ID,即考虑尚未合并的变化。在该第一搜索步骤i)中因此获得的数据记录ID被用作中间结果162,该中间结果162在随后的步骤ii)-步骤iv)中被进一步的数据丰富。进一步的数据可以包括在i)中确定的数据记录的数据值,并且可选地还包括作为在i)所确定的数据记录的上级或下级的进一步的数据记录。丰富数据作为结果164返回。

[0272] 在下文中,根据示例性实现变体描述了图3中一些数据结构的一些具体示例。

[0273] 逻辑数据记录各自由一个或多个字段数据值对组成,其中,数据值是映射ID。逻辑数据记录可以由任意数量的这种对组成。地址分配表226引用仅添加数据结构202内的AOD条目,该AOD条目按接收到相应的更改指令的顺序存储各种逻辑数据序列的各个字段的数据值。新AOD条目通常被写入到数据结构202的端部,其中,在生成新AOD条目的过程中,DMS确定新数据记录与先前数据记录的差异,并指定和存储新AOD条目的变化。新AOD条目包含指向先前内容的跳转地址(“指针”)。特别地,指针可以指定从数据结构202内的第一AOD条目开始计算的文件偏移量。一旦写入,AOD条目就不再更改,而只是添加。由于在每个逻辑数据记录中,根据一些实现变体,同一字段可能在逻辑数据记录中多次出现,并且可能被分配不同的映射ID,因此每个替换的字段数据值对优选地也被存储在否定列表中,而现在的当前字段数据值对被写入到肯定列表。首次将字段数据值对添加到数据记录时,新字段数据值配对将被添加到肯定列表。在删除的情况下,字段数据值对被添加到否定列表。肯定列表和否定列表例如关于图9被更详细地描述。为了在多次编辑的情况下优化系统加载,根据该方法的实施例,每个AOD条目都包含标识符(标志),该标识符(标志)指定当前AOD条目是否需要加载前置项(“不完整AOD条目”)(“完整的”或“完全加载的”AOD条目)。然而,所有更改请求的历史仍然可用,并且可以以防篡改的方式例如存储为区块链。

[0274] 附加地或替代地,AOD条目还可以包括其他标识符,例如指定AOD条目是否指存在上级和/或下级数据记录的数据记录的标识符(与所谓的“重复组”相关的标识符)。优选地,

至少一些逻辑数据记录包含相应的引用字段,例如“上级”和/或“下级”字段。此字段包含该字段的相关数据记录的上级或下级的数据记录的ID。如果要为引擎输入多个部件,则部件例如作为逻辑数据记录存储,并带有其产品编号和其技术特性,其中,“下级”字段中的部件数据记录包含引擎的数据记录ID。如果在写入引擎数据记录后创建部件条目,则通过仅由“下级”=TRUE标识符组成的补充AOD条目来写入对引擎的逻辑数据记录的更改。这可以例如通过存储该标识符的位值1来实现。现在,可以生成任意数量的新部件数据记录,并可以写入相应的AOD条目,其共同特征是字段数据值对“从属于”-“引擎MF-3000”。在数据输出期间,如果需要扩展或递归输出,则不仅可以输出特定部件,还可以输出关于安装或可能安装该特定部件的引擎类型的信息。该过程可以递归地应用,因为对于每个引擎,可能有包含该引擎的上级车辆。因此,可以在不事先知道数据结构的情况下写入和输出任意嵌套的内容。

[0275] 在一些实现示例中,仅添加数据结构还支持与逻辑数据记录的更改相关的完整时间历史的存储和重建。在下面的示例中,字段ID (FID) =2被保留用于历史信息的串联。例如,如果一个人的地址因搬家而发生变化,那么先前地址没有错,但其临时权限(“有效性”)已经过期。因此,它的字段相关的新数据值(新街道、门牌号、邮政编码、城镇)被转移到新的逻辑数据记录DSneu,在该逻辑数据记录DSneu中,先前数据记录的ID(DSalt因此是表示新数据记录的废弃版本的数据记录的ID)被输入到FId=2中。旧数据记录DSalt以AOD条目的形式存储有两个时间戳(从/到)。逻辑数据记录和与其对应的AOD条目的有效性时间段源于其数据有效的有效期。因此,该时间段的结束也自动地确定当前(或更多个当前)数据记录及其值中的至少一个值的有效性的开始。事件的数据记录历史(“历史条目”)的AOD条目的有效性的开始要么源于先前的有效性,要么可以手动设置。以下基于原始数据值的示例示出参照先前版本数据记录和分层链接数据记录的AOD条目。

[0276] Data record:

[0277] Company="Cortex Innovations"

[0278] is-employer-for="Employee A,1.2.1990-Employee B,7.12.1988-Employee C,1.5.2001

[0279] Offices="Isernhagen,Tischlerstr.1a,30916,Lower Saxony-Bendeleben until territorial reform:Kalkuferstr.7,99707 Bendeleben-Bendeleben after territorial reform:Kaluferstr.11,99703Kyfhäuserland

[0280] 这在仅添加文件中产生以下表示,其中,FID=1的字段包含返回对数据记录前置项版本的引用,其中,RID是数据记录ID的标识符,并且其中,标识符has_nR(“具有嵌套记录”)指示是否存在与AOD条目涉及的数据记录分层相关的进一步的数据记录:

[0281] RId:4711:Company Cortex Innovations Flag:has_nR

[0282] RId=4712,FId:1=4711 FId:12=First name-employee-A FId:13=Last name-employee-A,FId:4:1.2.1990

[0283] RId=4713,FId:1=4711 FId:12=First name-employee-B FId:13=Last name-employee-B,FId:4:7.12.1988

[0284] RId:4714,FId:1=4711,FId:12=First name-employee-C,FId:13=Last name-employee-C,4:1.5.2001

[0285] ...

[0286] RId:9876654,FId:1=4711,2:First name-employee-D,3:First name-last name-D,4:8.6.54

[0287] ...

[0288] RId:123456789FId:1=4711,FId:47=Isernhagen,FId:46=30916,FId:50=Tischlerstr.1a,FId:55=Lower Saxony

[0289] RId:1234568FId:1=4711,FId:47=Kyfhäuserland,FId:46=99703,FId:50=Kalkuferstr.11,FId:55=Thuringia

[0290] ...

[0291] RId=999991234FId:2=1234568,FId:47=Bendeleben,FId:FId:46=99706,FId:50=Kalkuferstr.7,T1=1,T2=30.05.2017

[0292] 该示例示出引用数据记录RId=999991234的AOD条目不仅包含该数据记录RId=999991234的ID,而且在FID=2的字段中还包含相同数据记录的前置项版本的RId,即1234568。如果要输出Cortex公司的数据记录的完整历史或仅先前有效的版本,DMS系统可以在地址分配表中搜索先前版本参考字段(FID=2)的内容,即值999991234。因此,它将被发送到包含该先前版本的最后更改的AOD条目的地址,以便可以输出该AOD条目中包含的数据,以及(如果适用)可以经由该AOD条目内的跳转地址到达的进一步的历史数据。

[0293] 图4示出不同类型的原始数据的示例,原始数据由DMS系统分解为原始数据值(在该情况下为Unicode字符的字符串),并分配给映射到映射ID的字段,映射ID用作逻辑数据记录的数据值。这些逻辑数据记录以字段特定数据值列表的形式存储,字段特定数据值列表实际上是字段特定映射ID列表。在下文中,为了说明的目的,最初获得的数据值被描述为逻辑数据记录的数据值,但是优选地,逻辑数据记录和字段特定数据值列表的数据值是这些原始数据值的数字映射ID。

[0294] 例如,数据结构302、304和306是JSON格式的制造商的引擎的产品数据表(由于空间原因,此处示出为制表符分隔的文本文件)。例如,将从原始数据中获得的逻辑数据记录导入数据库中的方式可以是,三个JSON文件302、304和306中的每一个都被解释为单独的数据对象或单独的逻辑数据记录,每个单独的数据对象或单独的逻辑数据记录都有数据记录ID。每个数据对象包含几个字段数据值对,例如字段“功率(Power)”的特定数据值、字段“扭矩(Torque)”的特定数据值等。

[0295] 下面示出数据结构是具有油漆经销商的油漆的各种特性规范的Excel表。每行308-313恰好包含一个数据对象(数据记录)。在解析期间,每个识别的数据对象可以被分配例如与该Excel表的标识符相结合的行号作为数据记录ID。

[0296] 一些原始数据可以以文本数据的形式提供,例如文本文件314-318。例如,纯语法解析器可以用于将该文本分解为单独的词语(每个词语都充当一个数据值)。例如,语法解析器可以是词法分析器(tokeniser),它将自然语言文本分解为充当数据值的词语(如果需要,除了一些停用词)。在这种情况下,词语/数据值可以例如在导入期间自动分配字段,例如通过NLP技术自动识别它们的语义含义。

[0297] 原始数据的另一部分可以例如以贸易登记摘录320-324的形式提供。这些可能包含键值字段和自由文本的混合。

[0298] 原始数据来自不同来源,但部分具有重叠内容(“Gelb-AG”),部分具有不明确的数

据值(“银”)。然而,实施例允许所有这些数据的有效集成和处理,同时解决语义歧义,例如通过在导入期间考虑原始数据的上下文,并根据上下文将数据值“银”解释为姓氏或金属,以及将其映射ID与相关联的数据记录ID一起存储在名字数据值列表中或金属数据值列表中,或者如果该映射ID已经存在于其中,则相应地扩展分配给该映射ID的数据记录ID的集合。

[0299] 图5示出将图4中所示的一些原始数据转换为字段特定数据值列表的示例。这里,在该方法的优选实现中,原始数据值的数值映射ID实际上存储在列表中,但是,为了说明的目的,这里示出具有原始数据值。

[0300] 在无冗余列表402、404、406、408、412、414、416中示出的数据值也仅仅是一种选择,通常列表相当长。

[0301] 根据本文所示的示例,由DMS系统生成和管理的所有字段特定数据值列表都是无冗余的,即它们每个数据值只包含一次。优选地,还对数据值进行排序,使得如果基于按顺序对搜索项和列表的已搜索数据值进行的排序,排除了列表内的进一步搜索将导致命中,则可以中止列表内的顺序搜索。

[0302] 例如,逻辑数据记录包括以下字段:颜色、制造商、(制造商的)油漆ID、姓氏和金属类型。这些字段分别由字段特定数据值列表402、404、406、408和412表示。

[0303] 根据上下文,可能会将相同的数据值“Gelb-AG”分配给不同的字段。例如,数据值“银”是一些逻辑数据记录中的一种颜色,并且存储在链接到包含值“银”作为颜色标识符的那些数据记录的数据记录ID的列表402中。数据值“银”可以被分配给其他逻辑数据记录中的字段“金属类型”,并且被存储在列表412中,该列表412链接到包含作为金属类型的值“银”的那些数据记录的数据记录ID。

[0304] 列表中数据值的存储取决于它们被分配到的各自字段,但与逻辑数据记录的隶属关系无关。因此,当生成列表时,原始数据中数据对象的原始结构在结构方面被完全分解,因为向数据记录分配数据值仍然存在,并且可以以数据记录ID的形式重建。

[0305] 优选地,数据结构还包括表示不同数据记录之间的“下级”关系类型的无冗余数据值列表414和/或表示不同数据记录之间的“上级”关系类型的无冗余数据值列表416。

[0306] 例如,列表414包含个体数据记录ID的非冗余、有序的列表,该列表可以由于有序而被快速搜索。这些数据记录ID存储在“键ID(Key ID)”列中。这些“键ID”中的每一个都存储为链接到数据记录的一个或更多个数据记录ID,一个或更多个数据记录ID是该“键ID”的上级。例如,列表414的键ID列中的ID可以表示可以被合并到一个或更多个较大部件(例如引擎)中的特定部件。例如,具有数据记录ID(RID) 304的部件可以并入MF-3000、MF-3020和MF-6000类型的引擎中。因此,这些引擎类型在某种程度上是这些部件的上级。

[0307] 例如,DMS系统可以接收对引擎部件的搜索请求,并且在步骤i)中,首先仅确定这些部件的数据记录ID。如果请求指定也要输出上级部件,则搜索具有这些部件的数据记录ID的列表414,以确定引擎类型(或其他上级部件)的一个或更多个数据记录ID,并且还完成并输出具有属性(数据值)的这些上级引擎或部件的数据记录ID。

[0308] 类似地,可以在请求中或者例如在DMS系统的配置文件中指定,除了对查询的搜索值的查询结果之外,还要返回从属于这些查询结果的数据记录。因此,例如也可以首先处理确定一个或更多个引擎类型的数据库查询,其中,从属于这些引擎类型的那些数据记录的

ID,例如因为它们表示这些引擎类型中的部件,也可以通过分析数据值列表416来自动确定。

[0309] 通过提供表示关于数据记录的排序或下级关系的关系的非冗余数据值列表414、416,可以有效地输出每个可想到的搜索查询的结果的大量上下文信息,而不需要指定复杂的并且因此通常非常慢的数据库查询。通过在最后414、416中迭代重复搜索,还可以输出3月2日或其他级别的上级和/或下级数据记录。因此,例如,在非常复杂的机器的情况下,可以针对每个部件非常有效地确定并输出其子部件直至单个螺杆的总数。

[0310] 因此,提供了一种用于通过分层关系链接的数据记录的高效搜索和聚合设施,该高效搜索和聚合设施也可以由具有较小RAM和/或计算能力的数据处理设备使用。

[0311] 图6示出用于使用数据库的合并和非合并版本的分布式系统500的框图。

[0312] 例如,系统500可以包括提供器计算机系统100和一个或多个接收器计算机系统506、508。接收器计算机系统用于在数据库104中以非冗余字段特定数据值列表的形式存储和保持当前逻辑数据记录。例如,根据一些实现变体,提供器计算机系统100可以包括DMS系统102,其可以经由API接收和处理搜索请求,和/或可以经由API耦接至各种源系统502、504,从这些源系统502、504接收、处理原始数据并将其存储为列表形式的逻辑数据记录。如已经描述的,数据库可以例如包括在特定时间点(合并时间点)处的静态数据库部分101。这尤其包括多个逻辑数据记录,其字段特定数据值以字段特定非冗余数据值列表的形式以分布式方式存储。对现有和已经合并的数据值和/或数据记录的更改最初不存储在这些字段特定数据值列表中,而是以仅添加数据结构内的AOD条目的形式存储,并且优选地以否定列表和/或肯定列表的形式存储。仅添加数据结构的尚未合并的部分以及否定列表和肯定列表表示数据库104的动态部分103。优选地,只有提供器计算机系统500包含用于导入新数据记录的合适程序,例如经由解析器118-130或能够从原始数据中提取逻辑数据记录的其他程序模块110。这确保了只有一个数据库实例被持续地修改和更新,并且同一数据库的不同实例不会被相互独立地修改并且因此可能变得不一致。

[0313] 然而,可能需要将数据库内容物理地复制到一个或多个接收方系统的本地存储器,例如,以确保接收器系统也可以访问数据库内容,而不管网络连接的可用性如何。为此目的,仅将数据库的静态部分101传送到接收器计算机系统506通常是足够的,例如如果接收器计算机系统不一定总是需要具有数据库104的最新版本,并且如果接收器计算机系统506上的本地副本在其自身内是一致的就足够了。

[0314] 根据系统500的一些示例,不仅静态部分被自动地从提供器计算机系统100复制到接收器计算机系统,而且动态部分也被自动地复制。这在图6中说明,例如,关于接收器计算机系统508。在这种情况下,接收器计算机系统508还包含模块510,用于将静态部分与动态部分中指定的变化合并,并且还用于在数据库搜索查询的处理过程中分析动态数据结构(肯定列表、否定列表、仅添加数据结构的未合并部分)。

[0315] 通常,各个计算机系统500、506、508、502、504经由网络(例如因特网)连接。

[0316] 然而,图6中所示的系统架构500只是许多可能的选项之一。许多替代架构是可能的。例如,可以在不同的数据处理系统上执行对原始数据的处理,例如存储数据库104并执行数据库查询和数据库合并。

[0317] 图7示出用于执行数据库查询的方法的流程图。该方法的步骤已经结合图1进行了

解释。

[0318] 图8示出在数据库查询中考虑尚未合并的数据值更改的方法的流程图。

[0319] 在步骤702中,DMS系统接收包含例如关于名字字段的搜索值Stefan的搜索查询。

[0320] 在下一步骤710中,DMS系统搜索映射表210以识别被分配给数据值或搜索值“Stefan”的映射ID之一。例如,这可以是数值“150”。此映射ID“150”在下文中用作实际搜索值。

[0321] 搜索值“150”用于在数据库的静态部分101上,即对字段特定数据值列表执行数据库搜索。在这里描述的示例中,搜索值150将用于搜索名字数据值列表,并且在图7所示的搜索方法的步骤i)中,将在静态数据上确定数据记录ID的集合,作为静态部分的搜索的结果返回。该结果表示在最近合并时间点的时间处数据库104的内容,并且因此不包括自该最近合并时间点以来获得的更改请求可能对结果产生的影响。例如,为了考虑尚未合并的更改,除了图7中描述的步骤之外,还执行图8中描述的步骤。

[0322] 在步骤706中,搜索数据库104的动态部分103的数据结构216,以确定包含名字字段的数据记录,在尚未合并的更改指令之后,该名字字段现在包含或不再包含数据值“Stefan”。特别地,这些数据结构216可以是包含数据值(特别是映射ID)的排序和可搜索阵列的数据结构。可搜索阵列可以是,例如,排序列表或搜索树,特别是B树。这种数据结构的示例如图9中所示。DMS系统被配置为将与单个数据值和/或整个数据记录相关的更改指令最初存储在仅添加数据结构中,并存储在包含肯定列表和否定列表的数据结构中,例如如图9中所示。在数据结构216中,分配给特定字段的数据值被表示为阵列的元素,该特定字段将根据尚未合并的更改指令之一被添加到一个或更多个数据记录或从一个或更多个数据记录中删除。这些数据值或元素中的每一个被分配被称为肯定列表的数据记录ID的列表和/或被称为否定列表的数据记录ID的列表。肯定列表选择性地包含那些数据记录的ID,由阵列的元素表示的数据值要被新添加到所述字段中。否定列表选择性地包含那些数据记录的ID,那些数据记录不应再包含由该字段中的阵列元素表示的数据值。

[0323] 在步骤706中,用搜索值“150”搜索数据结构216,直到找到表示该搜索值“150”的元素(例如B树的节点)为止,从而确定名字“Stefan”。此外,将确定分配给这些节点的肯定列表和否定列表。肯定列表包含用于根据由DMS系统在最近合并时间之后接收到的更改指令将数据值“Stefan”或“150”包含在“名字”字段中的所有数据记录ID的列表,例如因为具有该名字的新的个人数据记录要被添加到数据库和/或因为现有个人数据记录的名字已经更改(例如在拼写错误校正之后)。否定列表包含所有数据记录ID的列表,根据由DMS系统在最近合并时间之后接收到的更改指令,所有数据记录ID不应再包含先前数据值“Stefan”,例如因为该个人数据记录将被删除或仅名字发生了更改。

[0324] 在步骤708中,计算数据记录ID的最终结果列表,该最终结果列表考虑数据库104的静态部分和动态部分:在静态部分上确定的数据记录ID的集合,即搜索值150的名字数据值列表与包含在搜索值150的肯定列表中的数据记录ID的集合合并。因此,该并集现在还包含数据记录ID,根据最近且尚未合并的更改指令,该数据记录ID还应在“名字”字段中包含数据值150或Stefan。此外,形成该并集与包含在搜索值150的否定列表中的数据记录ID的集合的差集。以这种方式获得的差集不再包含在最近合并时间点之后被完全删除的那些数据记录的ID,或者不再包含名字150或Stefan。

[0325] 步骤710中计算的差集可以用作搜索步骤i)的新结果,与图7中的步骤612相比,该差集还考虑了动态的、尚未合并的变化。优选地,搜索现在还包括执行步骤614、616和618,类似于关于图7描述的步骤,以便不仅输出数据记录ID,而且输出具有数据值的完整数据记录,并且可选地还输出上级和/或下级数据记录或先前数据记录版本。

[0326] 图9示出具有多个肯定列表和否定列表的数据结构216的示例。例如,数据结构可以是字段特定数据结构,并且例如可以仅包含分配给特定字段(例如,名字)的数据值(优选地以映射ID的形式)。然而,也可以以这样的方式实现数据结构,例如,将数据库中所有数据值的整体存储在单个可搜索阵列中,其中,在这种情况下,阵列的每个元素被分配一个或多个字段(例如,数据值银可以表示金属或姓氏或颜色)。在这里描述的实现示例中,数据结构216仅表示“名字”字段的数据值或映射ID。

[0327] 图9中所示的可搜索阵列是B树,其中映射ID根据其数值进行排序存储。在该树中搜索值150的搜索在包含映射ID(MID)150的节点处结束。否定列表220和肯定列表218被分配给该节点。

[0328] 肯定列表218包含所有数据记录的ID,所有数据记录ID要根据在最近合并时间点之后接收到的更改指令而被新写入数据库中,并且所有数据记录ID在名字字段中包含名字Stefan或映射ID 150。肯定列表218另外包含所有数据记录的数据记录的ID,这些数据记录在最近合并时间点之前已经是数据库的一部分,但是其名字仅根据在最近合并时间点之后接收到的更改指令而被更改为Stefan。

[0329] 否定列表220包含根据在最近合并时间点之后接收到的更改指令要从数据库中删除的所有数据记录的ID,并且在名字字段中包含名字Stefan或映射ID 150。此外,否定列表包含在最近合并时间点之前已经是数据库的一部分并且在名字字段中包含名字Stefan或映射ID 150的所有数据记录的数据记录ID,但是根据在最近合并时间点之后接收到的更改指令,其将被更改为Stefan或150以外的值。

[0330] 图10示出合并前后的数据结构的框图,以及仅添加文件结构的读取加速的(独立于合并)优化的效果。

[0331] 例如,在合并数据库的过程中,所有肯定列表和否定列表的内容可以以字段特定数据值列表的更改的形式存储。由于肯定列表和否定列表是链接到字段标识符和数据值(例如,映射ID)来存储的,存储在否定列表或肯定列表中的每个数据记录ID都包含关于必须更改数据库的已经合并的静态部分的方式的信息,以便也反映包含在肯定列表和否定列表和/或在仅添加数据结构的尚未合并部分中的信息。合并仅包含已作为数据值包含在相应字段相关数据值列表中的数据值的新数据记录将不会更改字段特定数据值列表中的数据值或映射ID的数量(除非为隐藏出现频率的目的而为现有数据值生成了额外的映射ID)。但是,现有数据值或映射ID现在被附加地链接到新数据记录的ID。如果删除了数据记录,则会从包含该数据记录中包含的数据值或映射ID的所有字段特定数据值列表中删除该数据记录ID。即使进行了更改,例如覆盖数据值,在许多情况下,只有将数据记录ID分配给相应数据值或映射ID才会发生更改。只有当第一次将特定数据值分配给特定字段时,或者当覆盖或删除作为唯一包含该数据值的数据记录的数据记录的数据值时,合并才意味着字段特定数据值列表中包含的数据值或映射ID的数量变化。

[0332] 因此,在合并过程中,字段特定数据值列表116被转换为合并字段特定数据值列表

228.合并列表228包含数据值(或映射ID)和数据记录ID的分配,数据值(或映射ID)和数据记录ID以在最近的过去合并时间点和当前合并的时间点之间的更改指令的形式反映从DMS系统接收到的所有更改。更改指令可以是,例如,经典的基于SQL的删除(DELETE)、更新(UPDATE)或插入(INSERT)指令,或者不同语法的其他形式的写入指令。

[0333] 此外,具有数据值的可搜索阵列(例如,列表、搜索树)的数据结构被转移到这些数据结构的合并版本234中。数据结构的合并版本可以包含额外的元素(排序列表的列表元素、搜索树的节点等),这些元素反映了先前未包含在数据库中的新添加的数据值或映射ID。如果已经从数据集中完全删除了一些数据值或映射ID,则数据结构216的合并版本234也可以包含较少数量的元素。特别地,合并数据结构234的肯定列表和否定列表被清空。这确保了肯定列表和否定列表仅包含与根据在当前合并时间点之后接收的更改指令要更改的那些数据记录相关的数据记录ID。

[0334] 仅添加数据结构202被连续地更新。因此,它在合并过程中通常不会被清空。

[0335] 然而,优选地,DMS系统被配置为在合并过程中或独立于合并对仅添加数据结构202进行更改,这在步骤216和218的执行过程中大大加速了对该数据结构的评估。这将仅添加数据结构202转换为仅添加数据结构的优化形式236。优化由DMS系统生成并存储在仅添加数据结构202中的AOD条目组成,AOD条目被称为“完整的”或“完全加载的”AOD条目,并且AOD条目将当前数据值(特别是当前映射ID)分配给数据记录的每个字段。这些“完整的”AOD条目不是响应于接收到更改请求而写入的,而是独立的。例如,完整AOD条目可以在预定义时间间隔之后生成并存储,在合并过程中,响应于来自用户的命令,或者响应于DMS系统确定在执行搜索查询的过程中仅添加数据结构202的处理花费太长时间,例如超过所需的预定义时间最大值。

[0336] 通常,AOD条目只包含根据相应的更改请求进行更改的一个或几个字段的当前数据值。它们并不包含更改后的数据记录的所有数据值。为了确定所有数据值,DMS系统必须遵循各个AOD条目中包含的地址,每个地址都指的是最后写入的AOD条目,该最后写入的AOD条目指向相同的数据记录。

[0337] AOD条目尤其以这样的方式生成,即DMS系统在写入要更改的特定数据记录的新AOD条目期间已经用最新/最近的更改指令确定了已经存在的AOD条目的地址。这种确定可以使用地址分配表非常快速和高效地执行,其中,数据记录ID仍然被存储为链接到现有AOD条目的地址。然后,将具有该数据记录的最近更改的新AOD条目编写到仅添加数据结构202中,其中,该新AOD条目包含确定的现有的、最近AOD条目的地址。此外,必须更新地址分配表中分配给数据记录ID的AOD条目地址,以便数据记录ID现在指向新AOD条目的地址。

[0338] 因此,通过追踪包含在仅添加数据结构202内的AOD条目中的地址,可以快速地搜索与特定数据记录相关的所有AOD条目的整个序列,直到与该数据记录相关的仅添加数据结构202中的第一AOD元素。

[0339] 尽管在仅添加数据结构202内的这种基于地址的跳跃是非常有效的,但是通过在偶尔的优化过程中也写入完整AOD条目,可以进一步提高效率,该优化可以与其他数据库合并同步地或异步地运行,从而可以在几个跟踪步骤之后终止跟踪,因为当达到完整AOD条目时,数据记录的所有当前有效的数据值都是已知的。

[0340] 根据一个实现变体,AOD数据结构在合并过程中或独立于合并进行优化,以便加速

步骤216和步骤218的执行。这可以例如通过创建AOD文件的内容的副本来实现,其中,该副本组合了引用相同数据记录的几个AOD条目的内容,从而创建了AOD数据结构,其中AOD条目都是完整AOD条目,或者其至少具有比原始AOD数据结构更高的完整AOD条目内容。修改后的副本然后用于完成要输出的数据记录。当处理AOD条目以完成结果数据记录时,需要对AOD条目地址进行较少的访问,因为AOD条目中的更高比例是完整AOD条目,因此不需要访问过去的进一步的数据记录。

[0341] 图11示出根据另一个实现示例的用于完整逻辑结果数据记录的有效输出的数据结构。数据结构包括地址分配表226和仅添加数据结构202,如已经针对其他实现变体所描述的,例如图3中所示的变体。逻辑数据记录908在这里被示为具有虚线的框,以澄清逻辑数据记录优选地以非冗余数据值列表或映射ID列表的形式而不是以表形式存储。标识符“Distr.Stor.”表示这些数据记录的字段值或映射ID的分布式存储发生在几个非冗余数据值列表中。

[0342] 在图11所示的实现变体中,DMS系统被配置为在创建和/或导入新逻辑数据记录时,还在地址分配表226中创建唯一分配给该数据记录的新条目,并且为新逻辑数据记录分配ID,该ID显式地或隐式地指定分配给该数据记录的地址分配表中的条目的存储器地址。因此,数据记录ID“#109”指定地址分配表226内唯一分配给该数据记录的行的存储器地址,而另一数据记录的数据记录ID“#110”指定地址分配表226内唯一分配给该另一数据记录的行的存储器地址。这由双箭头表示。数据记录#110的地址分配表226中的条目/行包含特定AOD条目(即在字段F3的值被设置为“b”的过程中具有最近更改的条目)的存储器地址。该AOD条目依次引用/包含下一个较旧AOD条目的地址,该下一个较旧AOD条目引用相同的数据记录#110,在这种情况下,该数据记录指定字段F3的较早值,即“a”。

[0343] 优选地,数据库查询分两步执行:在第一步中,仅确定要返回的逻辑数据记录的ID,而不确定这些数据记录的字段值。在这里描述的变体中,该步骤可以通过评估数据值列表来执行。

[0344] 仅在第二步中,从数据记录的ID开始,使用地址分配表和AOD数据结构来确定当前分配给结果数据记录的数据值(例如映射ID),并且返回包括数据值的完整数据记录。在该第二步中,优选地直接执行对数据结构226和202中的各个条目的访问,即在各个数据结构中没有搜索步骤,其中,第二步包括:a)访问在第一步中确定的数据记录ID中显式地或隐式地指定的地址分配表中的条目的地址;该步骤用于确定包含在这些地址分配表条目中的AOD地址;b)访问在a)中确定的AOD条目地址,以便确定要返回的数据记录的字段值中的最新更改;以及c)如果b)中涉及的AOD条目不完整,则评估对包含在每个AOD条目中的下一个较旧AOD条目的地址的引用,以直接访问和评估下一个较旧AOD条目,并且重复步骤c),直到已经为要返回的数据记录的所有字段确定了当前字段值为止。如果字段值是映射ID,则可以在进一步的步骤中访问映射表,以将映射ID替换为原始数据值,并返回包含原始数据值而不是映射ID的数据记录。

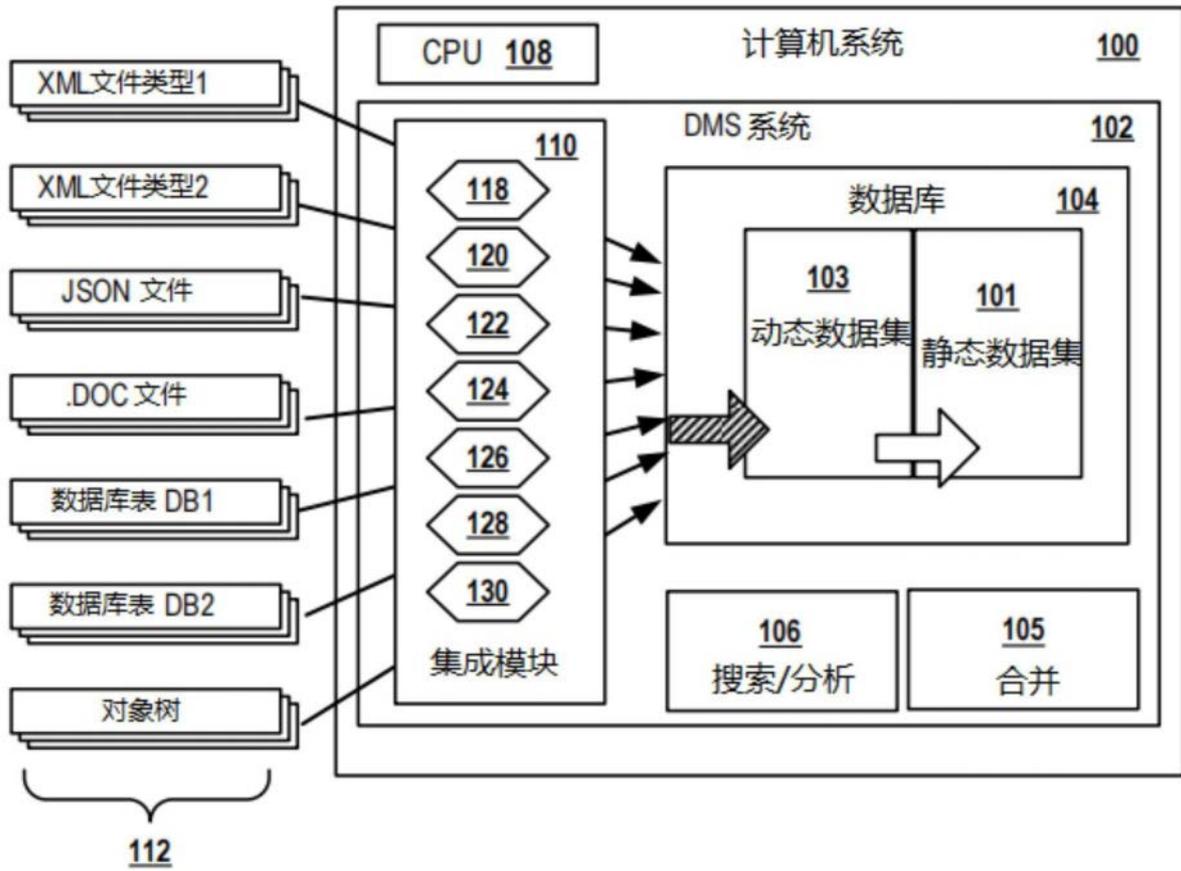


图1

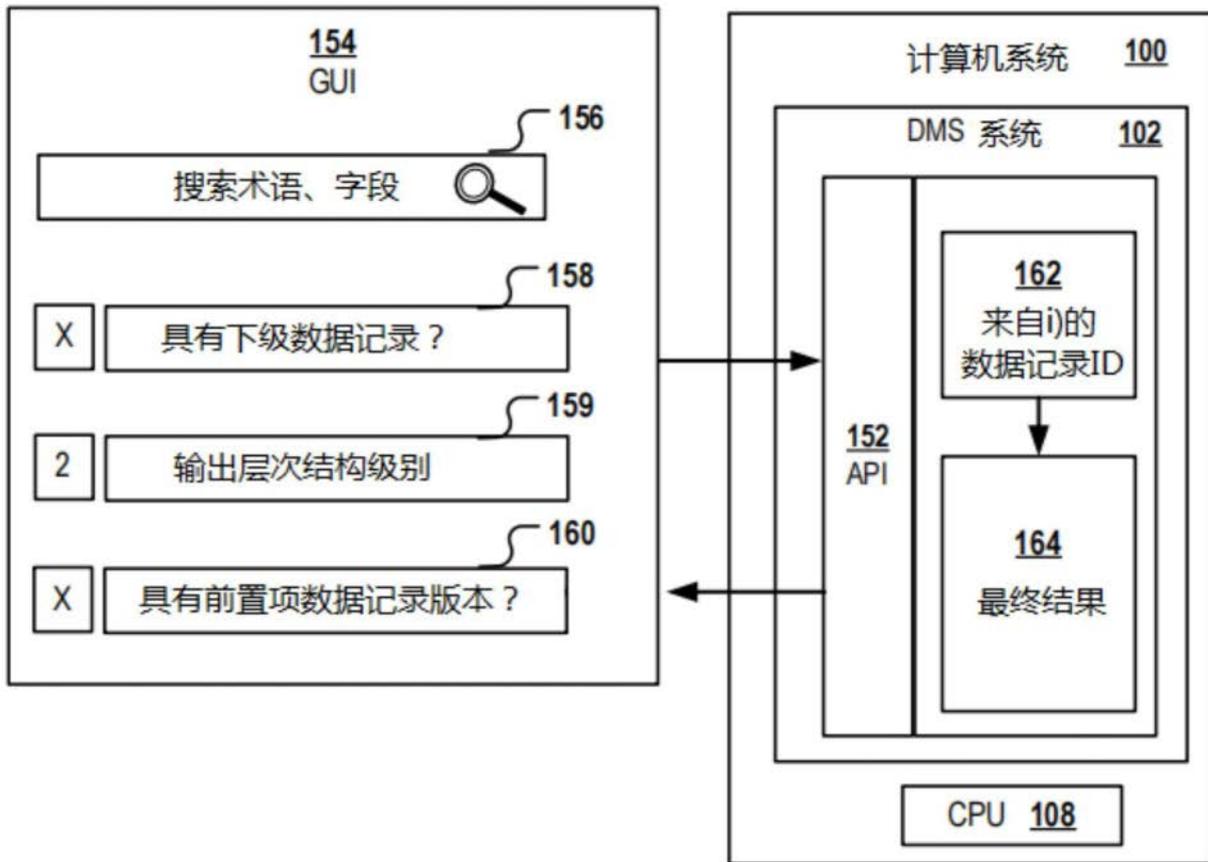


图2

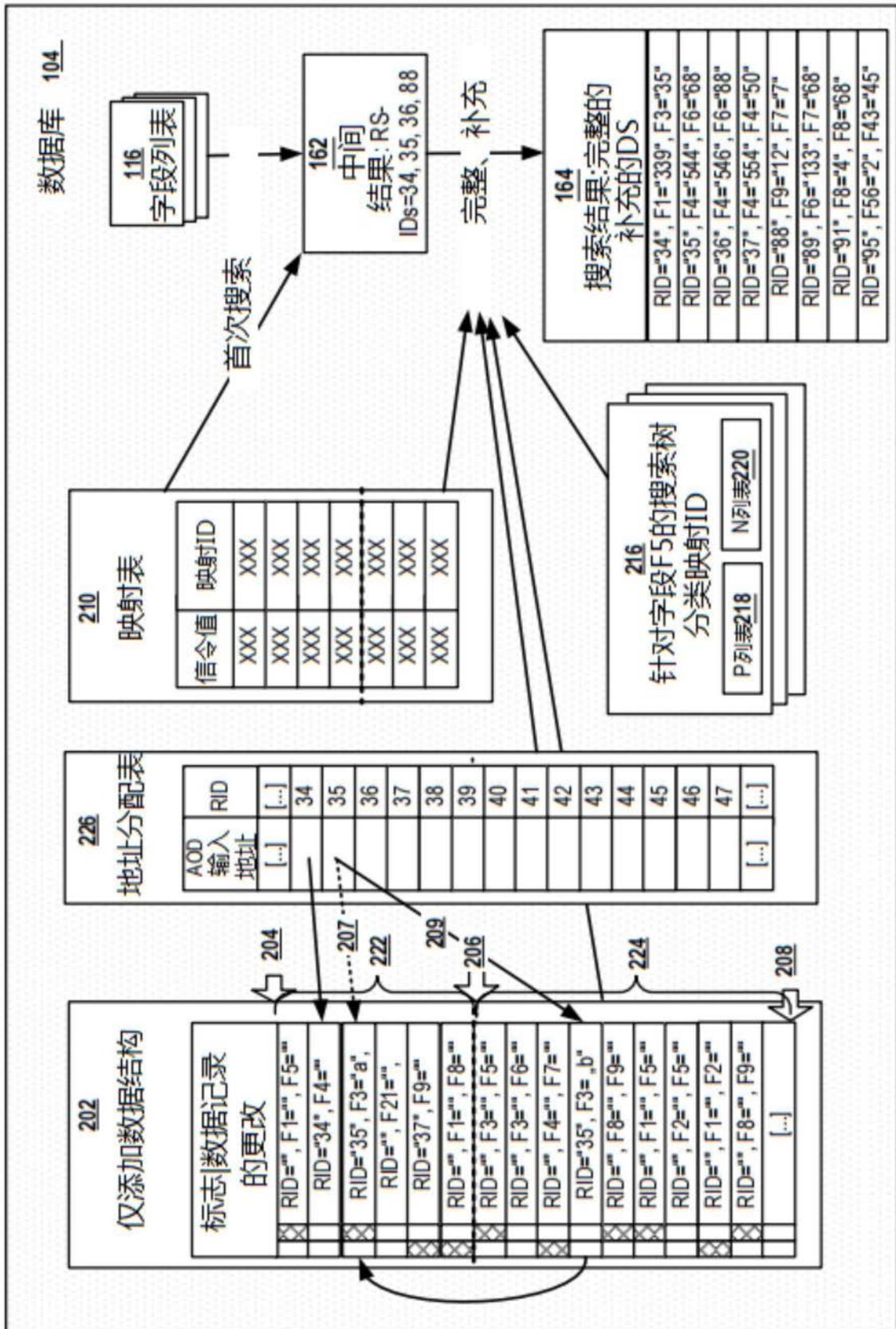


图3

引擎的产品数据表 (JSON)

产品数据表:	引擎 CL 28378-RS-A123	302	
类型:	三相引擎		304
功率:	4 kW		306
扭矩:	(Nm) 0.43/0.42		
速度:	(1/min) 1340/1608		
制造商:	Gelb AG		
制造地点:	Produktionsstraße 3, 16321 Bernau		
颜色:	银		
油漆饰面:	闪亮		
可安装在:	MF-3000, MF-3020, MF-3100, MF-6000		

产品数据表油漆 (.xsl)

#	油漆ID	油漆类型	颜色	指令
[...]	[...]	[...]	[...]	[...]
308	L-467-HT7	亮光漆	红	表面必须被清洁 ...
310	L-467-HT8	亮光漆	银	表面必须被清洁 ...
312	L-467-HT9	亚光漆	黄	表面必须被清洁 ...
313	L-467-HT23	亚光漆	银	表面必须被清洁 ...
[...]	[...]	[...]	[...]	[...]

消息文本 (.txt)

Gelb-AG 在2005财政年度通过销售引擎获利3000万欧元。贵金属交易，尤其是金和银，产生了400万欧元的利润。该公司由Paul Schmidt于1978年创立	314	
[...]		316
		318

贸易登记摘录 (.txt)

公司:	Gelb-AG	320	
总部:	Kastanienallee 34, 56185 Wiesbaden		322
成立于:	02.12.1978		324
董事(名、姓):	马丁·西尔柏(Martin Silber)		
创始人:	保罗·施密特(Paul Schmidt)		
描述:	Gelb-AG是引擎、变速器和机动车部件的制造商。		

图4



图5

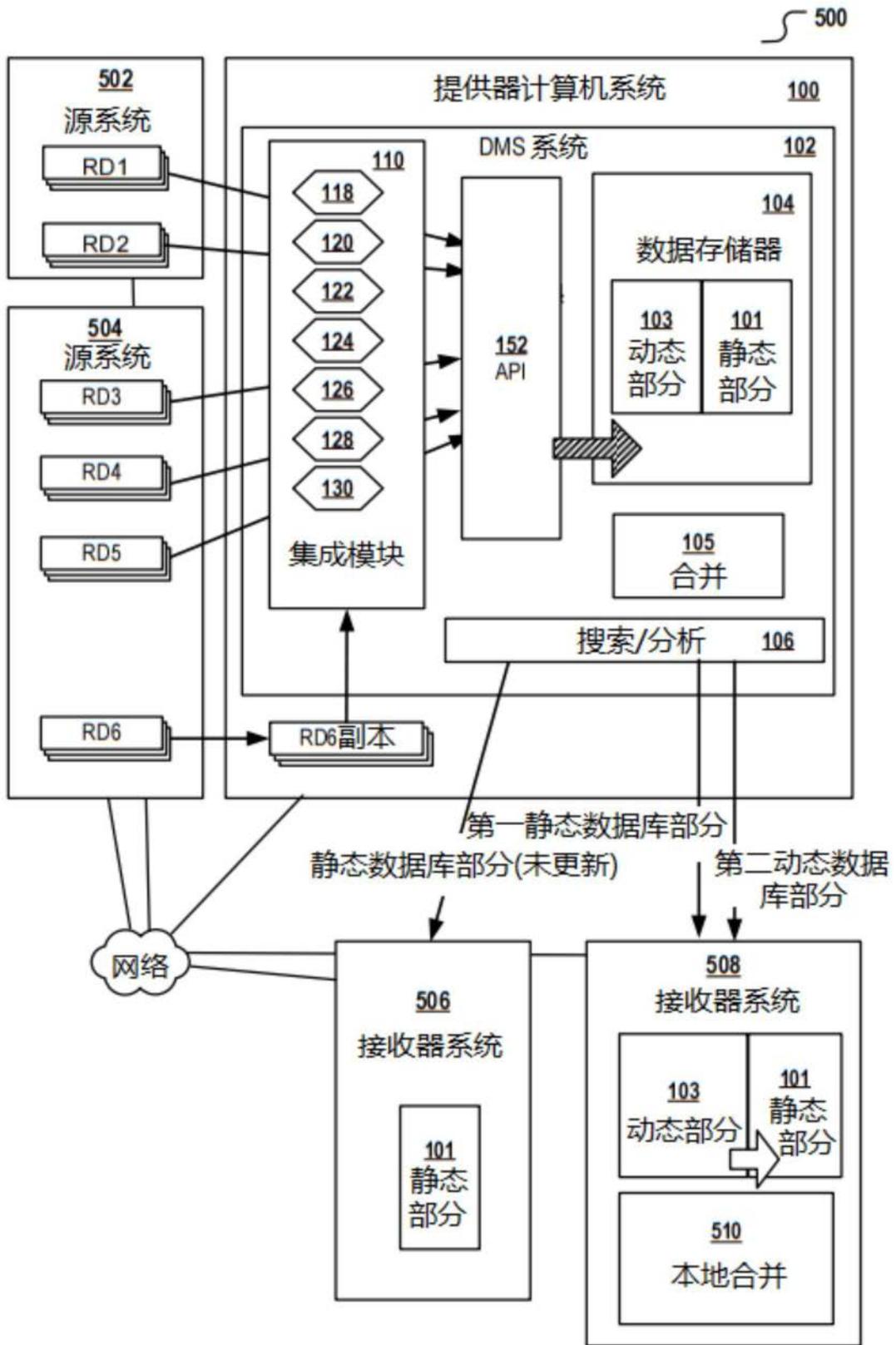


图6

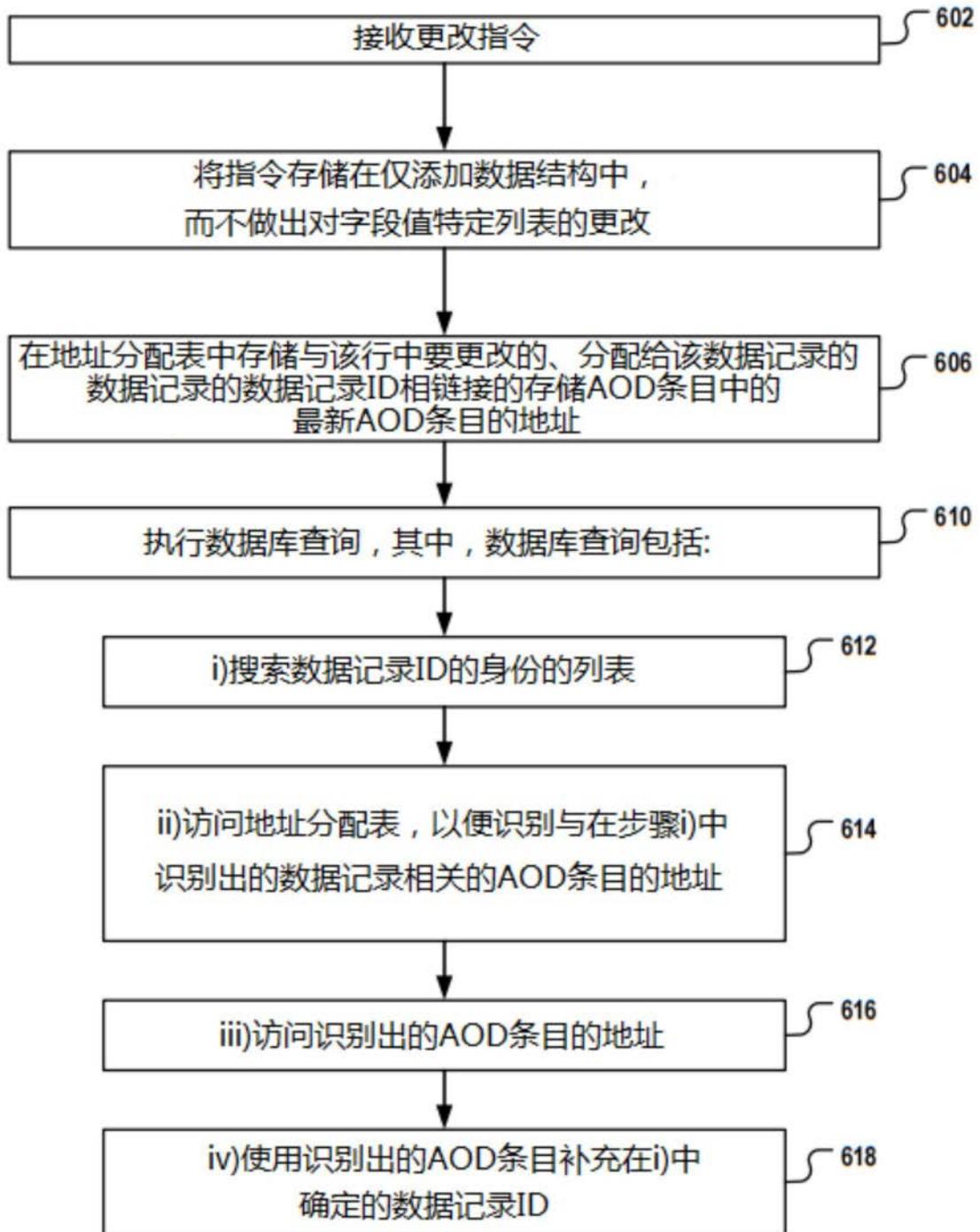


图7

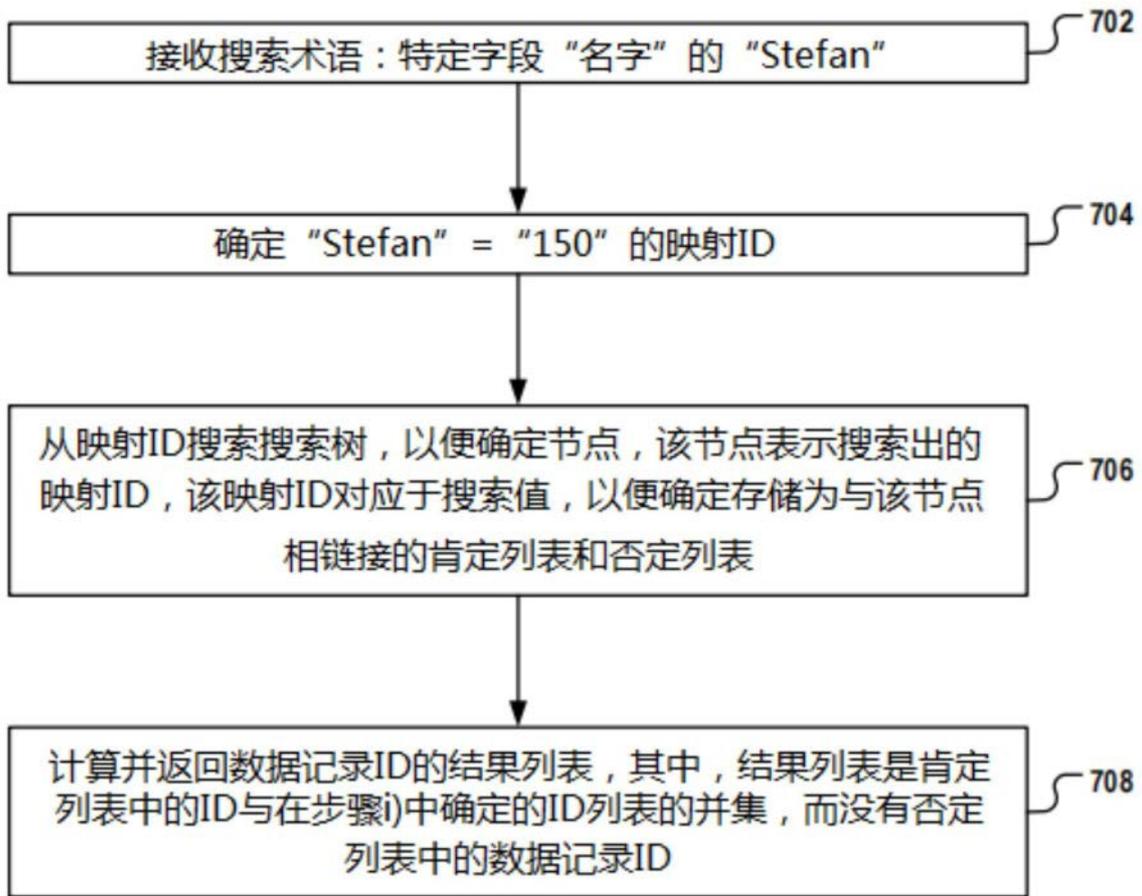


图8

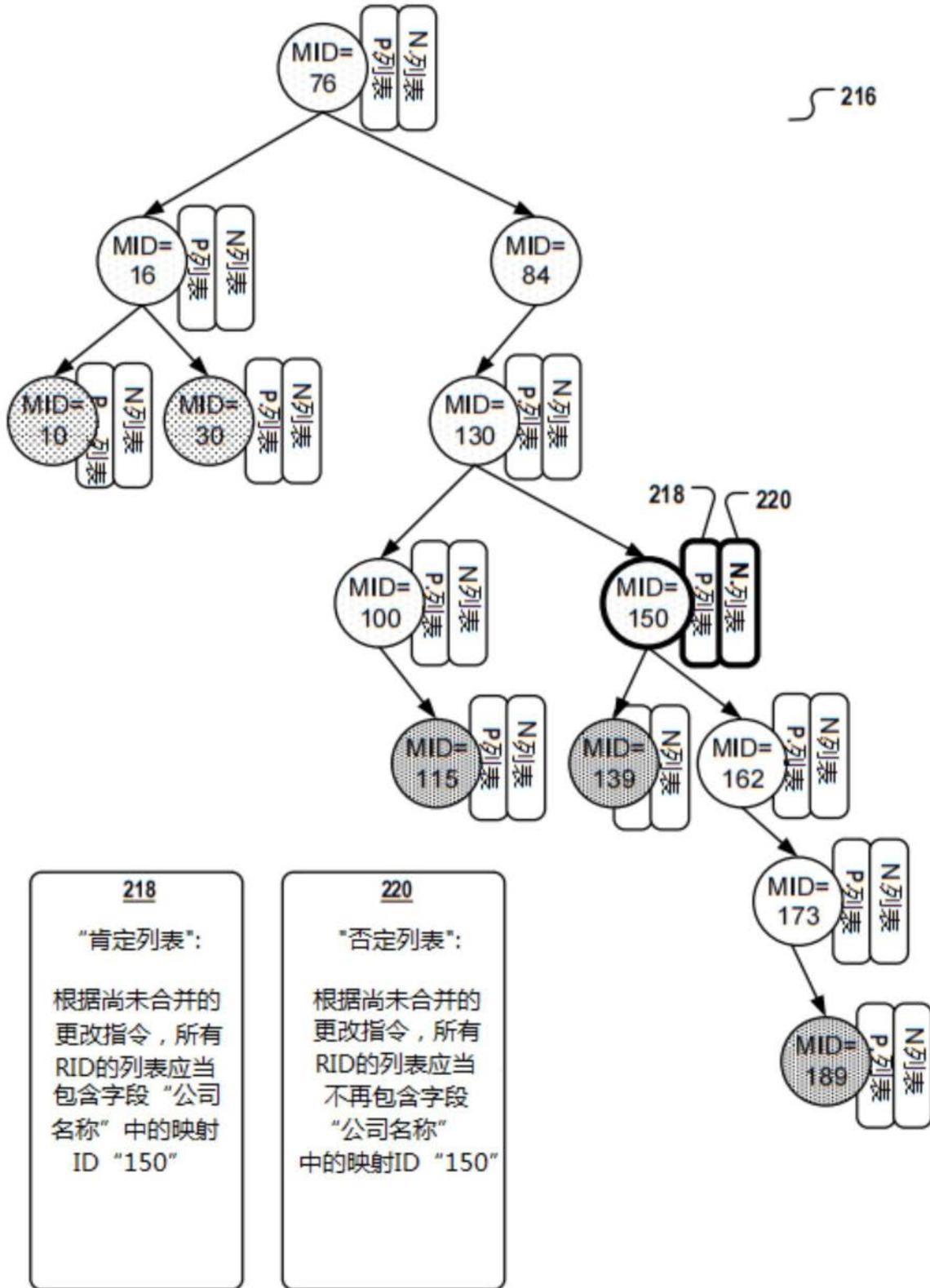


图9

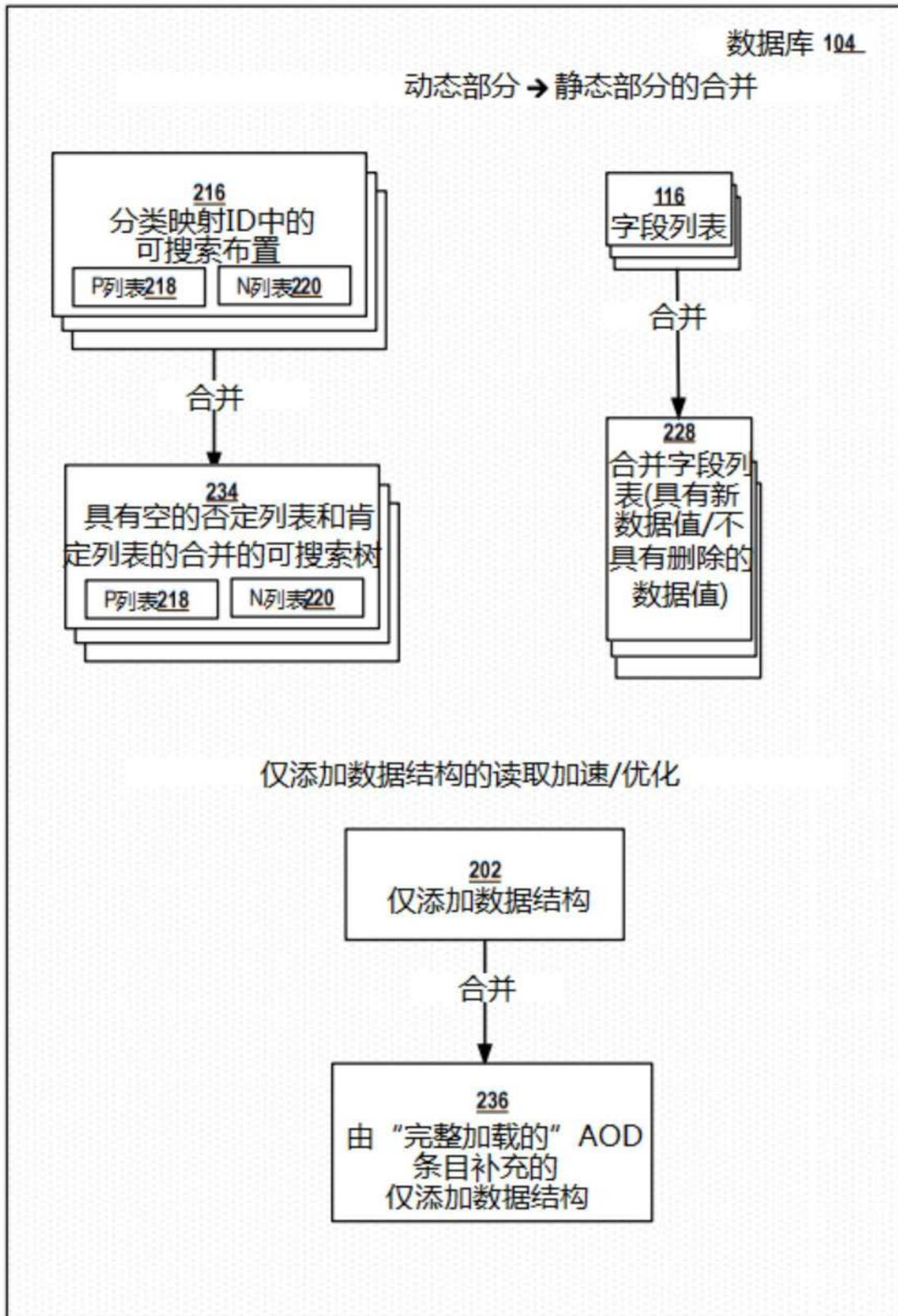


图10

