

数据库访问方法及装置

技术领域

[01] 本说明书一个或多个实施例涉及计算机技术领域，尤其涉及一种数据库访问方法及装置。

5 背景技术

[02] 为了响应用户的数据库访问请求，服务器在接收到用户的数据库访问请求后，通常会针对该访问请求，制定相应的访问策略和方案。此处的访问策略和方案可以被称为执行计划。之后，通过执行制定的执行计划来实现对数据库的访问，并向用户返回所希望的结果。

- 10 [03] 需要说明的是，对上述执行计划的制定，在理想的情况下，可以是为每个访问请求都制定对应的执行计划。然而在制定执行计划的过程中，通常需要对数据库中的数据执行统计、采样等繁琐的操作，也即制定过程比较耗时，这通常会使用户的访问时间变的过长。因此，为了优化用户的访问时间，目前的执行计划的制定方式主要包括两种：
- 15 第一种，对同一类型的访问请求，执行一次制定执行计划的过程，也即同一类型的访问请求共享一个执行计划。第二种，根据访问请求对应的访问语句中包含的参数不同，制定不同的执行计划。总之，通过不同方式制定不同的执行计划，会对数据库的访问造成不同的影响。

发明内容

- 20 [04] 本说明书一个或多个实施例描述了一种数据库访问方法及装置，可以提高用户对数据库访问的效率和准确性。

[05] 第一方面，提供了一种数据库访问方法，包括：

[06] 接收数据库访问语句；

[07] 解析所述数据库访问语句，确定所述数据库访问语句中包含的目标用户参数；

[08] 确定所述数据库访问语句所属的语句类型；

- 25 [09] 获取预先存储的与所述语句类型的数据库访问语句对应的执行计划分组；所述执行计划分组中的各个执行计划通过不同的用户参数相区分；

- [10] 根据所述目标用户参数，从所述执行计划分组中选取目标执行计划；
- [11] 执行所述目标执行计划。
- [12] 第二方面，提供了一种数据库访问装置，包括：
- [13] 接收单元，用于接收数据库访问语句；
- 5 [14] 解析单元，用于解析所述接收单元接收的所述数据库访问语句，确定所述数据库访问语句中包含的目标用户参数；
- [15] 确定单元，用于确定所述数据库访问语句所属的语句类型；
- [16] 获取单元，用于获取预先存储的与所述确定单元确定的所述语句类型的数据库访问语句对应的执行计划分组；所述执行计划分组中的各个执行计划通过不同的用户参数
- 10 相区分；
- [17] 选取单元，用于根据所述目标用户参数，从所述获取单元获取的所述执行计划分组中选取目标执行计划；
- [18] 执行单元，用于执行所述选取单元选取的所述目标执行计划。
- [19] 本说明书一个或多个实施例提供的数据库访问方法及装置，接收数据库访问语句。
- 15 解析数据库访问语句，以确定该数据库访问语句中包含的目标用户参数。确定该数据库访问语句所属的语句类型。获取预先存储的与该语句类型的数据库访问语句对应的执行计划分组。根据上述目标用户参数，从获取的执行计划分组中选取目标执行计划。执行目标执行计划。也即在本说明书实施例中，语句类型相同的数据库访问语句共享一组执行计划，而同一组执行计划中的各个执行计划又通过不同的用户参数相区分，由此，可
- 20 以提高用户对数据库访问的准确性。此外，还可以大大减少制定的执行计划的数量。在执行计划减少的情况下，可以提高用户对数据库的访问效率，并且可以达到节约计算机资源的目的。

附图说明

- [20] 为了更清楚地说明本说明书实施例的技术方案，下面将对实施例描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图仅仅是本说明书的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其它的附图。
- 25

[21] 图 1 为本说明书提供的数据库访问方法的应用场景示意图;

[22] 图 2 为本说明提供的对象类型分组的确定方法流程图;

[23] 图 3 为本说明书一个实施例提供的数据库访问方法流程图;

[24] 图 4 为本说明书提供的计划空间的探索过程示意图;

5 [25] 图 5 为本说明书另一个实施例提供的数据库访问方法流程图;

[26] 图 6 为本说明书一个实施例提供的数据库访问装置示意图。

具体实施方式

[27] 下面结合附图, 对本说明书提供的方案进行描述。

10 [28] 本说明书提供的数据库访问方法可以应用于如图 1 所示的场景中, 图 1 中, 客户端可以向服务器发送数据库访问请求, 该数据库访问请求可以通过数据库访问语句(如, 结构化查询语言(Structured Query Language, SQL)来实现。图 1 中的服务器可以包括计划缓存, 该计划缓存可以是指从服务器的内存划分得到的一块存储空间, 用于存储服务器生成的执行计划。此处的执行计划可以是由服务器根据上述数据库访问语句生成的。此外, 服务器还可以对数据库进行访问。具体地, 服务器是通过对执行计划的执行来实
15 现对数据库的访问。

[29] 图 1 中的数据库可以包括多个对象, 该对象可以是指表、视图等可能存在于数据库访问语句中的访问实体。在本说明书中, 以对象为表为例来说。

[30] 需要说明的是, 在执行本说明书提供的数据库访问方法之前, 可以先对数据库中的对象进行分组, 从而确定多个对象类型分组。在一种实现方式中, 该多个对象类型分
20 组的确定方法可以如图 2 所示。图 2 中, 该确定方法可以包括:

[31] 步骤 210, 从数据库中选取对象模式相同的多个对象。

[32] 以表为例来说, 上述对象模式相同的多个对象可以是指列的数据类型相同且索引模式相同的多张表。其中, 索引模式相同可以理解为各张表所建立的索引个数以及建立索引的列相同。

25 [33] 举例来说, 对于通过如下的 SQL 语句建立的两张表: user_01 和 user_02, 则认为该两张表的对象模式相同。

[34] create table user_01 (id integer primary key, gmt_create date, balance_01 number(10,2)

index user_01_gmt_blance on (gmt_create, blance_01));

[35] create table user_02 (id integer primary key, gmt_create date, balance_02 number(10,2)
index user_02_gmt_blance on (gmt_create,blance_02));

[36] 通过上述 SQL 语句可以得出, user_01 总共包括 3 列, 且该 3 列的数据类型分别为:

- 5 “integer”、“date”以及“number(10,2)”。此外, user_01 还建立了索引, 且建立索引的列为第 3 列。同理, 可以得出, user_02 总共包括 3 列, 且该 3 列的数据类型分别为: “integer”、“date”以及“number(10,2)”。此外, user_02 还建立了索引, 且建立索引的列为第 3 列。由此, 可以认为 user_01 和 user_02 是对象模式相同的两张表。

[37] 步骤 220, 对该多个对象进行分组。

- 10 [38] 需要说明的是, 在本说明书中, 对象的分组是一个渐进的过程。如, 对步骤 210 中选取的多个对象, 可以先针对其中任意两个对象进行分组, 即计算该两个对象之间的相似度。当计算的相似度小于预设阈值 d 时, 将该两个对象划分为一个分组。当计算的相似度不小于预设阈值 d 时, 将该两个对象分别划分为两个分组。之后, 针对多个对象中的第三个对象进行分组, 如, 在前述两个对象划分为一个分组时, 可以计算第三个对象与其中一个对象之间的相似度。若计算的相似度小于预设阈值 d, 则将第三个对象也划分到上述一个分组中。若计算的相似度不小于预设阈值 d, 则计算第三个对象与另一个对象之间的相似度。若计算的相似度小于预设阈值 d, 则将第三个对象也划分到上述一个分组中; 否则, 将第三个对象划分到另一个分组中。

- 20 [39] 在前述两个对象分别划分为两个分组时, 可以计算第三个对象与其中一个对象之间的相似度。若计算的相似度小于预设阈值 d, 则将第三个对象划分到该一个对象所在分组中。若计算的相似度不小于预设阈值 d, 则计算第三个对象与另一个对象之间的相似度。若计算的相似度小于预设阈值 d, 则将第三个对象划分到该另一个对象所在分组中; 否则, 将第三个对象划分到其它分组中。以此类推, 直至将上述多个对象均划分到对应的分组中。

- 25 [40] 总之, 本说明书提供的对象分组方法是针对每个对象依次进行的。即对某个对象进行分组时, 可以将该对象依次与已经存在的多个分组内的多个对象进行相似度比较 (即计算相似度), 当与任意一组内的某的对象之间的相似度满足要求的时候, 即停止比较, 并将该对象划分到该分组中; 否则, 继续比较。如果最终没有任何一个对象与该对象“相似” (即相似度小于预设阈值 d), 则将该对象单独划分为一组, 之后对其它对象继续执行上述相似度比较过程。
- 30

- [41] 在一种实现方式中，可以通过计算两个对象之间的欧几里得距离来确定上述相似度。以计算两张表的相似度为例来说，可以预先设定三个属性：行数 r ，唯一值个数 ndv 以及平均行长 rl 。之后，在计算任意两张表之间的相似度时，可以先确定该两张表的上述三个属性的属性值。最后，基于该两张表各自的三个属性的属性值，来计算两者之间的相似度。在一个例子中，任意两张表之间的相似度计算公式可以如下所示。

$$similarity = \sqrt{(r_1 - r_2)^2 + (ndv_1 - ndv_2)^2 + (rl_1 - rl_2)^2} \quad (\text{公式 1})$$

- [42] 其中， $similarity$ 为两张表之间的相似度， r_1 、 ndv_1 以及 rl_1 分别为其中一张表的行数、唯一值个数以及平均行长， r_2 、 ndv_2 以及 rl_2 分别为另一张表的行数、唯一值个数以及平均行长。
- 10 [43] 当然，在实际应用中，也可以定义其它的属性，如，列数等，本说明书对此不作限定。
- [44] 可以理解的是，上述步骤 210-步骤 220 可以是重复执行的，直至数据库中的每个对象均被划分到对应的分组。在将数据库的每个对象均被划分到对应的分组之后，可以将划分得到的各个分组作为多个对象类型分组。
- 15 [45] 步骤 230，建立数据库中的各个对象与多个对象类型分组之间的对应关系。
- [46] 在一个例子中，上述对应关系可以如表 1 所示。

表 1

对象	对象类型分组
user_01	group_01
user_02	group_01
dept_01	group_02
...	...

- [47] 表 1 中， $user_01$ 和 $user_02$ 属于同一对象类型分组 ($group_01$)，也即 $user_01$ 与 $user_02$ 的对象模式相同且两者之间的相似度小于预设阈值 d 。
- 20 [48] 本说明书通过将对象模式相同且相似度小于预设阈值 d 的对象划分到同一的对象类型分组，可以在分库分表的场景下，聚合同一逻辑表的多张物理分表，从而可以大大减少生成的执行计划的数量，这可以达到节约计算机资源的目的。
- [49] 需要说明的是，对上述多个对象类型分组，可以周期性自动检测每个对象类型分

组中各个对象的对象模式是否相同，并在对象模式相同时，针对每个对象，计算该对象与所属对象类型分组中其它对象之间的相似度。当该对象与其它对象之间的相似度均不小于预设阈值时，重新确定该对象所属的对象类型分组。由此，来确保数据库中各个对象分组的准确性，进而可以提供数据库访问的准确性。此外，上述自动检测对象分组准确性的过程，可以减少对用户的依赖和侵入，从而增加了数据库系统的稳定性。

[50] 还需要说明的是，本说明书基于对象模式和相似度确定对象类型分组的策略，可以避免由用户指定对象类型分组而使数据库的稳定性和易用性差的问题。

[51] 可以理解的是，在将数据库中的各个对象划分到对应的对象类型分组之后，就可以执行本说明书提供的数据库访问方法。

10 [52] 图 3 为本说明书一个实施例提供的数据库访问方法流程图。所述方法的执行主体可以为具有处理能力的设备：服务器或者系统或者装置，如，图 1 中的服务器。如图 3 所示，所述方法具体可以包括：

[53] 步骤 310，接收数据库访问语句。

15 [54] 此处，可以是由服务器接收客户端发送的数据库访问语句。该数据库访问语句可以是指 SQL 语句。如，可以为：select * from user_01 a, dept_01 b where a.id = b.id and a.id = 1。

[55] 步骤 320，解析数据库访问语句，确定数据库访问语句中包含的目标用户参数。

20 [56] 在数据库访问语句为 SQL 语句时，可以是根据 SQL 语句的语法，来解析该 SQL 语句。上述目标用户参数可以是指 SQL 语句中的固定值。其可以包括：整数、浮点数以及字符串等。

[57] 如前述例子，该 SQL 语句中包含的目标用户参数为：1。

[58] 步骤 330，确定数据库访问语句所属的语句类型。

25 [59] 在一种实现方式中，可以先对数据库访问语句进行参数化。之后根据参数化后的数据库访问语句，确定所属的语句类型。其中，对数据库访问语句进行参数化的步骤可以包括：

[60] 步骤 a，从数据库访问语句中提取对应的对象集合。

[61] 其中，对象集合中的对象可以是指上述表、视图等可能存在于数据库访问语句中的访问实体。在数据库访问语句为 SQL 语句时，可以通过 SQL 语句的语法对 SQL 语

句进行解析，来提取上述对象集合。

[62] 以数据库访问语句为: `select * from user_01 a, dept_01 b where a.id = b.id and a.id = 1` 为例来说，提取的对象集合可以为: `O={ user_01,dept_01}`。

5 [63] 步骤 b，对对象集合中的对象，确定对象所属的对象类型分组，并获取对应的分组名称。

[64] 在预先确定多个对象类型分组，并建立数据库中的各个对象与多个对象类型分组之间的对应关系时，可以根据该对应关系，来确定对象集合中的对象所属的对象类型分组。如前述例子，可以根据表 1 所示的内容，确定 `user_01` 所属的对象类型分组为: `group_01`，`dept_01` 所属的对象类型分组为: `group_02`。

10 [65] 步骤 c，将数据库访问语句中的对象替换为对应的分组名称，并将目标用户参数替换为通配符。

[66] 此处的通配符可以是指任何与数据库访问语句中其它字符或者字符串不相同的一个或多个字符。该字符可以为“@”、“?”或者“%”等特殊字符。以通配符为:“?”为例来说，对前述例子，将 `user_01` 替换为 `group_01`，将 `dept_01` 替换为 `group_02`，将 15 1 替换为“?”之后，该 SQL 语句转化为字符串: `select * from group_01, group_02 where group_01.id = group_02.id and group_01.id = ?`。

[67] 可以理解的是，上述字符串即为该数据库访问语句所属的语句类型。

[68] 步骤 340，获取预先存储的与该语句类型的数据库访问语句对应的执行计划分组。

20 [69] 在一种实现方式中，可以从服务器的计划缓存中获取上述执行计划分组。该计划缓存中可以预先存储有多个与不同的语句类型对应的执行计划分组，其中，每个执行计划分组通过对应的哈希值进行标识，执行计划分组中的各个执行计划通过不同的用户参数相区分。

[70] 在一个例子中，计划缓存的内容可以如表 2 所示。

表 2

hash 值	语句类型	执行计划分组		
XXXX	select * from group_01, group_02 where group_01.id = group_02.id and group_01.id = ?	分组 1	plan1	1
			plan2	2
			plan3	3
YYYY	select * from group_01, group_03 where group_01.id = group_03.id and group_01.id = ?	分组 2	plan4	≤ 10
			plan5	> 10
ZZZZ	select * from group_01, group_02 where group_01.id = group_02.id and group_01.age = ?	分组 3	plan6	≤ 18
			plan7	(18,50)
			plan8	≥ 50
...	...	...		

[71] 当然，在实际应用中，上述计划缓存也可以不包括语句类型，或者还可以包括其它内容，本说明书对此不作限定。

[72] 具体地，在确定数据库访问语句所属的语句类型之后。可以根据预设的哈希函数，
5 计算所属语句类型对应的哈希值。之后根据计算的哈希值，获取对应的执行计划分组。

[73] 需要说明的是，计划缓存中的执行计划分组可以是由服务器在响应用户的数据库访问请求的过程中不断生成并存储的。上述执行计划分组的生成过程也可以称为计划空间的探索过程。该计划空间的探索过程可以如图 4 所示。

[74] 图 4 中，p1 和 p2 分别为参数空间中两个不同的点，用于表示两个不同的用户参数。
10 当服务器接收到包含有：p1 的某语句类型的数据库访问语句时，可以根据该 p1 生成相应的执行计划。该执行计划的生成过程可以示意为在图 4 所示的参数空间中划分出一块区域，该划分出的区域也可以称为计划空间。之后，当服务器接收到包含有 p2 的该语句类型的数据库访问语句时，可以计算 p2 与 p1 之间的距离 Δp （如，可以为欧几里得距离）。假设 Δp 大于阈值距离 D，可以根据 p2 生成相应的执行计划，即在图 4 所示的
15 参数空间中划分出另一个计划空间。需要说明的是，当服务器某次接收到的该语句类型的数据库访问中包含的用户参数：pi 与 p1（或者 p2）的距离 Δp 小于 D 时，不再生成新的执行计划。也即与 p1（或者 p2）共享执行计划。可以理解的是，随着用户参数的增加，整个参数空间将逐渐被划分出多个不同的计划空间。当部分计划空间出现重叠时，

即将两个计划空间合并为一个。最终，整个参数空间会被完全划分，计划空间的探索过程即认为结束。可以理解的是，在计划空间的探索过程结束之后，生成的多个执行计划即构成了一个执行计划分组。

5 [75] 当然，在实际应用中，p1 和 p2 也可以分别为参数空间中两个不同的数值范围，本说明书对此不复赘述。

[76] 需要说明的是，图 4 只是示出了在一维空间下，计划空间的探索过程。当数据库访问语句中同时包含多个不同的用户参数时，如，同时包含 n 个不同的用户参数时，该参数空间即构成了 n 维空间。在 n 维空间中，计划空间的探索过程类似。如，可以将 n 个不同的用户参数表示为数组：(a1,a2,a3,...,an)。之后在划分计划空间时，假设参数空间中已经为某个点 (b1,b2,b3,...,bn) 划分了相应的计划空间，则可以根据如下公式来计算上述数组与该点之间的距离 Δp 。

$$\Delta p = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + (a_3 - b_3)^2 + \dots + (a_n - b_n)^2} \quad (\text{公式 2})$$

[77] 之后，可以根据公式 2 计算的距离 Δp 与阈值距离之间的大小关系，来确定是否划分新的计划空间。其确定方法同上所述，在此不复赘述。

15 [78] 还需要说明的是，图 4 只是给出了一组执行计划分组的生成过程，也即服务器接收到的数据库访问语句的语句类型相同。当服务器接收到多种不同语句类型的数据库访问语句时，可以为每种语句类型均生成对应的执行计划分组。从而可以得到多组执行计划分组，具体可以参见表 2 所示的内容。

20 [79] 本说明书通过上述计划空间的探索过程生成执行计划的方法，可以大大提高执行计划的质量，从而可以进一步提高用户对数据库的访问效率。此外，本说明书针对一类类型的数据库访问语句生成执行计划分组的方式，可以避免大量的重复计算，从而可以加速执行计划的生成速度和效率。再者，通过该方式还可以大大减小生成的执行计划的数量，这可以降低对空间的占用，从而可以达到节约计算机资源的目的。

[80] 步骤 350，根据目标用户参数，从执行计划分组中选取目标执行计划。

25 [81] 可选地，为了提高选取的执行计划的质量，在执行步骤 350 之前，可以再次判断对象集合中的各个对象所属的对象类型分组是否发生了变化。在未发生变化的情况下，才执行步骤 350。

[82] 上述判断的过程可以为：对对象集合中的每个对象，判断该对象与所属对象类型

分组中各个对象的对象模式是否相同。如果对象模式相同，则依次计算该对象与各个对象之间的相似度。举例来说，假设对象集合为：{ user_01,dept_01}，且预先确定的数据库中的对象与对象类型分组之间的对应关系如表 1 所示。则可以确定 user_01 与 user_02 的对象模式是否相同。如果对象模式相同，可以计算该两者之间的相似度。此外，还可以确定 dept_01 与 group_02 中其它对象的对象模式是否相同。如果对象模式相同，则可以依次计算 dept_01 与 group_02 中其它对象之间的相似度。此处，相似度的计算可以参见公式 1，在此不复赘述。当各个对象与所属对象类型分组中任一对象之间的相似度小于预设阈值时，认为各个对象所属的对象类型分组未发生变化。需要说明的是，当对象模式不相同或者当对象模式相同但与所属对象类型分组中各个对象之间的相似度均不小于预设阈值时，可以重新确定所属的对象类型分组。如前述例子，假设 user_01 与 user_02 的对象模式不相同，或者假设对象模式相同，但两者之间的相似度不小于预设阈值，而 dept_01 与 group_02 中其它对象的对象模式相同且与组内某个对象之间的相似度小于预设阈值，则可以重新确定 user_01 所属的对象类型分组。

[83] 可以理解的是，在重新确定某个或者某几个对象所属的对象类型分组时，则可以重新执行上述步骤 330-步骤 340。

[84] 需要说明的是，对于步骤 350 中的执行计划分组，当如上所述的计划空间探索完成时，该执行计划分组中包括了所有的执行计划。从而根据目标用户参数，可以选取到对应的目标执行计划。即计算目标用户参数与执行计划分组中的各个执行计划对应的用户参数之间的欧几里得距离。当与任一执行计划对应的用户参数之间的欧几里得距离小于阈值距离时，将该任一执行计划作为目标执行计划。

[85] 当如上所述的计划空间探索还未完成时，目标执行计划的选取过程同上所述。不同的是，由于计划空间还未探索完成，因此可能存在选取不到目标执行计划的情况。也即可能会存在目标用户参数与执行计划分组中已生成的各个执行计划对应的用户参数之间的欧几里得距离均不小于阈值距离的情况。此时，可以继续上述计划空间的探索过程，即在图 4 所示的参数空间中划分出相应的计划空间，也即根据目标用户参数，生成新执行计划。从而执行计划分组中加入了新执行计划。

[86] 步骤 360，执行目标执行计划。

[87] 此处，如果选取到目标执行计划，则执行该目标执行计划。如果未选取到目标执行计划，则执行新执行计划。

[88] 综上,通过本说明书实施例提供的方案,既可以提高用户对数据库访问的准确性,又可以提高访问效率。

[89] 图 5 为本说明书另一实施例提供的数据库访问方法流程图。如图 5 所示,该方法可以包括:

- 5 [90] 步骤 510,接收数据库访问语句。
- [91] 步骤 520,解析数据库访问语句,确定数据库访问语句中包含的目标用户参数。
- [92] 步骤 530,从数据库访问语句中提取对应的对象集合。
- [93] 步骤 540,对对象集合中的对象,确定该对象所属的对象类型分组。
- [94] 步骤 550,根据确定的对象类型分组,对数据库访问语句进行参数化。
- 10 [95] 步骤 560,根据参数化后的数据库访问语句,确定该数据库访问语句所属的语句类型。
- [96] 步骤 570,获取预先存储的与该语句类型的数据库访问语句对应的执行计划分组。
- [97] 步骤 580,对对象集合中的对象,确定该对象与所属对象类型分组中各个对象的对象模式是否相同;若否,则执行步骤 590;若是,则执行步骤 5100。
- 15 [98] 步骤 590,重新确定该对象所属对象类型分组,并跳转至步骤 550。
- [99] 步骤 5100,依次计算该对象与上述各个对象之间的相似度。
- [100] 步骤 5110,判断与任一对象之间的相似度是否小于预设阈值。若否,则执行步骤 590;若是,则执行步骤 5120。
- [101] 步骤 5120,根据目标用户参数,从执行计划分组中选取目标执行计划。
- 20 [102] 步骤 5130,判断是否选取到目标执行计划;若是,则执行步骤 5140;若否,则执行步骤 5150。
- [103] 步骤 5140,执行目标执行计划。
- [104] 步骤 5150,根据目标用户参数,生成新执行计划。
- [105] 步骤 5160,执行新执行计划,并将新执行计划加入执行计划分组。
- 25 [106] 本说明书上述实施例首先确定数据库访问语句中对象所属的对象类型分组,之后再选取或者生成执行计划的方式,可以聚合对象类型相同的多个对象(如,聚合同一张

逻辑表的多张物理分表)，由此，可以达到执行计划共享的目的。

[107]与上述用户意图识别方法对应地，本说明书一个实施例还提供的一种用户意图识别装置，如图6所示，该装置包括：

[108]接收单元601，用于接收数据库访问语句。

- 5 [109]解析单元602，用于解析接收单元601接收的数据库访问语句，确定数据库访问语句中包含的目标用户参数。

[110]确定单元603，用于确定数据库访问语句所属的语句类型。

[111]确定单元603具体可以用于：

[112]对数据库访问语句进行参数化。

- 10 [113]根据参数化后的数据库访问语句，确定所属的语句类型。

[114]其中，对数据库访问语句进行参数化可以包括：

[115]从数据库访问语句中提取对应的对象集合。

[116]对对象集合中的对象，确定对象所属的对象类型分组，并获取对应的分组名称。

- 15 [117]将数据库访问语句中的对象替换为对应的分组名称，并将目标用户参数替换为通配符。

[118]获取单元604，用于获取预先存储的与确定单元603确定的语句类型的数据库访问语句对应的执行计划分组；该执行计划分组中的各个执行计划通过不同的用户参数相区分。

- 20 [119]选取单元605，用于根据目标用户参数，从获取单元604获取的执行计划分组中选取目标执行计划。

[120]执行单元606，用于执行选取单元605选取的目标执行计划。

[121]可选地，该装置还可以包括：分组单元607和建立单元608。

[122]选取单元605，还用于从数据库中选取对象模式相同的多个对象。

[123]分组单元607，用于对所述选取单元605选取的多个对象进行分组。

- 25 [124]执行单元606，还用于重复执行上述选取以及分组的步骤，直至数据库中的对象均被划分到对应的分组。

- [125] 建立单元 608, 用于将分组单元 607 得到的多个分组作为多个对象类型分组, 并建立数据库中的各个对象与多个对象类型分组之间的对应关系。
- [126] 确定单元 603 还具体可以用于:
- [127] 根据对应关系, 确定对象所属的对象类型分组。
- 5 [128] 可选地, 选取单元 605 具体可以用于:
- [129] 对对象集合中的对象, 确定对象与所属对象类型分组中各个对象的对象模式是否相同。
- [130] 如果对象模式相同, 则依次计算对象与各个对象之间的相似度。
- [131] 当与任一对象之间的相似度小于预设阈值时, 根据目标用户参数, 从执行计划分
- 10 组中选取目标执行计划。
- [132] 当与各个对象之间的相似度均不小于预设阈值时, 重新确定对象所属的对象类型分组, 并根据重新确定的对象类型分组, 获取对应的目标执行计划。
- [133] 可选地, 该装置还可以包括:
- [134] 计算单元 609, 用于对各个对象类型分组中的每个对象, 周期性依次计算该对象与
- 15 所属对象类型分组中其它对象之间的相似度。
- [135] 确定单元 603, 还用于当与其它对象之间的相似度均不小于预设阈值时, 重新确定该对象所属的对象类型分组。
- [136] 可选地, 选取单元 605 还具体可以用于:
- [137] 计算目标用户参数与执行计划分组中的各个执行计划对应的用户参数之间的欧几
- 20 里得距离。
- [138] 当与任一执行计划对应的用户参数之间的欧几里得距离小于阈值距离时, 将任一执行计划作为目标执行计划。
- [139] 可选地, 该装置还可以包括:
- [140] 生成单元 6010, 用于根据目标用户参数, 生成新执行计划。
- 25 [141] 加入单元 6011, 用于将生成单元 6010 生成的新执行计划加入执行计划分组, 并执行新执行计划。
- [142] 本说明书上述实施例装置的各功能模块的功能, 可以通过上述方法实施例的各步

骤来实现，因此，本说明书一个实施例提供的装置的具体工作过程，在此不复赘述。

- [143] 本说明书一个实施例提供的数据库访问装置，接收单元 601 接收数据库访问语句。解析单元 602 解析数据库访问语句，确定数据库访问语句中包含的目标用户参数。确定单元 603 确定数据库访问语句所属的语句类型。获取单元 604 获取预先存储的与确定的语句类型的数据库访问语句对应的执行计划分组。选取单元 605 根据目标用户参数，从执行计划分组中选取目标执行计划。执行单元 606 执行目标执行计划。由此，既可以提高用户对数据库访问的准确性，又可以提高访问效率。

[144] 需要说明是，本说明书实施例提供的数据库访问装置可以为图 1 中服务器的一个模块或者单元。

- 10 [145] 本领域技术人员应该可以意识到，在上述一个或多个示例中，本说明书所描述的功能可以用硬件、软件、固件或它们的任意组合来实现。当使用软件实现时，可以将这些功能存储在计算机可读介质中或者作为计算机可读介质上的一个或多个指令或代码进行传输。

- 15 [146] 以上所述的具体实施方式，对本说明书的目的、技术方案和有益效果进行了进一步详细说明，所应理解的是，以上所述仅为本说明书的具体实施方式而已，并不用于限定本说明书的保护范围，凡在本说明书的技术方案的基础之上，所做的任何修改、等同替换、改进等，均应包括在本说明书的保护范围之内。

权利要求书

- 1、一种数据库访问方法，其特征在于，包括：
接收数据库访问语句；
解析所述数据库访问语句，确定所述数据库访问语句中包含的目标用户参数；
5 确定所述数据库访问语句所属的语句类型；
获取预先存储的与所述语句类型的数据库访问语句对应的执行计划分组；所述执行计划分组中的各个执行计划通过不同的用户参数相区分；
根据所述目标用户参数，从所述执行计划分组中选取目标执行计划；
执行所述目标执行计划。
- 10 2、根据权利要求 1 所述的方法，其特征在于，所述确定所述数据库访问语句所属的语句类型，包括：
对所述数据库访问语句进行参数化；
根据参数化后的数据库访问语句，确定所属的语句类型；
其中，所述对所述数据库访问语句进行参数化，包括：
15 从所述数据库访问语句中提取对应的对象集合；
对所述对象集合中的对象，确定所述对象所属的对象类型分组，并获取对应的分组名称；
将所述数据库访问语句中的对象替换为对应的分组名称，并将所述目标用户参数替换为通配符。
- 20 3、根据权利要求 2 所述的方法，其特征在于，还包括确定多个对象类型分组的步骤，包括：
从数据库中选取对象模式相同的多个对象；
对所述多个对象进行分组；
重复执行上述选取以及分组的步骤，直至所述数据库中的对象均被划分到对应的分
25 组；
将划分得到的所述多个分组作为所述多个对象类型分组，并建立所述数据库中的各个对象与所述多个对象类型分组之间的对应关系；
所述确定所述对象所属的对象类型分组，包括：
根据所述对应关系，确定所述对象所属的对象类型分组。
- 30 4、根据权利要求 2 或 3 所述的方法，其特征在于，所述根据所述目标用户参数，从所述执行计划分组中选取目标执行计划，包括：

对所述对象集中的对象，确定所述对象与所属对象类型分组中各个对象的对象模式是否相同；

如果对象模式相同，则依次计算所述对象与所述各个对象之间的相似度；

当与任一对象之间的相似度小于预设阈值时，根据所述目标用户参数，从所述执行

5 计划分组中选取目标执行计划；

当与所述各个对象之间的相似度均不小于预设阈值时，重新确定所述对象所属的对象类型分组，并根据重新确定的对象类型分组，获取对应的目标执行计划。

5、根据权利要求 3 所述的方法，其特征在于，还包括：

10 对各个对象类型分组中的每个对象，周期性依次计算所述对象与所属对象类型分组中其它对象之间的相似度；

当与其它对象之间的相似度均不小于所述预设阈值时，重新确定所述对象所属的对象类型分组。

6、根据权利要求 1 所述的方法，其特征在于，所述根据所述目标用户参数，从所述执行计划分组中选取目标执行计划，包括：

15 计算所述目标用户参数与所述执行计划分组中的各个执行计划对应的用户参数之间的欧几里得距离；

当与任一执行计划对应的用户参数之间的欧几里得距离小于阈值距离时，将所述任一执行计划作为所述目标执行计划。

20 7、根据权利要求 1 所述的方法，其特征在于，当未选取到所述目标执行计划时，还包括：

根据所述目标用户参数，生成新执行计划；

将所述新执行计划加入所述执行计划分组，并执行所述新执行计划。

8、一种数据库访问装置，其特征在于，包括：

接收单元，用于接收数据库访问语句；

25 解析单元，用于解析所述接收单元接收的所述数据库访问语句，确定所述数据库访问语句中包含的目标用户参数；

确定单元，用于确定所述数据库访问语句所属的语句类型；

获取单元，用于获取预先存储的与所述确定单元确定的所述语句类型的数据库访问语句对应的执行计划分组；所述执行计划分组中的各个执行计划通过不同的用户参数相

30 区分；

选取单元，用于根据所述目标用户参数，从所述获取单元获取的所述执行计划分组

中选取目标执行计划;

执行单元, 用于执行所述选取单元选取的所述目标执行计划。

9、根据权利要求 8 所述的装置, 其特征在于, 所述确定单元具体用于:

对所述数据库访问语句进行参数化;

5 根据参数化后的数据库访问语句, 确定所属的语句类型;

其中, 所述对所述数据库访问语句进行参数化, 包括:

从所述数据库访问语句中提取对应的对象集合;

对所述对象集合中的对象, 确定所述对象所属的对象类型分组, 并获取对应的分组名称;

10 将所述数据库访问语句中的对象替换为对应的分组名称, 并将所述目标用户参数替换为通配符。

10、根据权利要求 9 所述的装置, 其特征在于, 还包括: 分组单元和建立单元;

所述选取单元, 还用于从数据库中选取对象模式相同的多个对象;

所述分组单元, 用于对所述选取单元选取的所述多个对象进行分组;

15 所述执行单元, 还用于重复执行上述选取以及分组的步骤, 直至所述数据库中的对象均被划分到对应的分组;

所述建立单元, 用于将所述分组单元得到的所述多个分组作为所述多个对象类型分组, 并建立所述数据库中的各个对象与所述多个对象类型分组之间的对应关系;

所述确定单元还具体用于:

20 根据所述对应关系, 确定所述对象所属的对象类型分组。

11、根据权利要求 8 或 9 所述的装置, 其特征在于, 所述选取单元具体用于:

对所述对象集合中的对象, 确定所述对象与所属对象类型分组中各个对象的对象模式是否相同;

如果对象模式相同, 则依次计算所述对象与所述各个对象之间的相似度;

25 当与任一对象之间的相似度小于预设阈值时, 根据所述目标用户参数, 从所述执行计划分组中选取目标执行计划;

当与所述各个对象之间的相似度均不小于预设阈值时, 重新确定所述对象所属的对象类型分组, 并根据重新确定的对象类型分组, 获取对应的目标执行计划。

12、根据权利要求 10 所述的装置, 其特征在于, 还包括:

30 计算单元, 用于对各个对象类型分组中的每个对象, 周期性依次计算所述对象与所属对象类型分组中其它对象之间的相似度;

所述确定单元，还用于当与其它对象之间的相似度均不小于所述预设阈值时，重新确定所述对象所属的对象类型分组。

13、根据权利要求 8 所述的装置，其特征在于，所述选取单元还具体用于：

计算所述目标用户参数与所述执行计划分组中的各个执行计划对应的用户参数之

5 间的欧几里得距离；

当与任一执行计划对应的用户参数之间的欧几里得距离小于阈值距离时，将所述任一执行计划作为所述目标执行计划。

14、根据权利要求 8 所述的装置，其特征在于，还包括：

生成单元，用于根据所述目标用户参数，生成新执行计划；

10 加入单元，用于将所述生成单元生成的所述新执行计划加入所述执行计划分组，并执行所述新执行计划。